Michel CROQUET

PC et Robotique



20 MONTAGES A FAIRE SOI-MEME POUR OUVRIR SON PC SUR LE MONDE EXTERIEUR

Techniques d'interfaçage



ETSF
EDITIONS TECHNIQUES ET SCIENTIFIQUES FRANÇAISES

Michel CROQUET

PC

et Robotique

Techniques d'interfaçage



Cet ouvrage a été vendu par la société ELECTROME
sous le nom de *Dossier PC et robotique*

ISBN 2 10 023986 4

DOSSIER PC ET ROBOTIQUE

Traduction en Turbo Pascal V6.0 de Borland des programmes écrits en GW BASIC de Microsoft du Dossier PC et Robotique.

CARTE 8 SORTIES

Programme LED1.PAS

Ce programme permet le clignotement de 2 leds connectées sur la carte 8 sorties.

CARTE 1 AFFICHEUR / CARTE 8 SORTIES

Programme 7SEG1.PAS

Ce programme permet l'allumage d'un segment dont on compose la valeur sur le clavier de l'ordinateur.

Programme 7SEG2.PAS

Ce programme affiche le chiffre composé au clavier (de 0 à 9), sur la carte 1 afficheur.

Programme 7SEG3.PAS

Ce programme affiche les chiffres de 0 à 9 suivis du point en permanence avec une vitesse de défilement ajustable par les touches + et - du clavier de l'ordinateur.

CARTE CANIL / CARTE 8 SORTIES

Programme CANIL1S.PAS

Ce programme permet de saisir au clavier la valeur binaire codée décimale des leds de la carte CANIL à allumer.

Programme CANIL3S.PAS

Ce programme permet de composer 20 séquences lumineuses s'enchaînant l'une derrière l'autre.

CARTE CANIL / CARTE PIA/PC

Programme CANIL1.PAS

Ce programme permet de vérifier, grâce à l'allumage des leds, les valeurs transmises par l'interface PIA/PC.

Programme CANIL2.PAS

Identique à CANIL1.PAS, mais pour composer un nouveau code BCD, il faut appuyer sur le poussoir de la carte.

Programme CANIL3.PAS

Ce programme permet de composer 20 séquences s'enchaînant l'une derrière l'autre.

CARTE TRIAC / CARTE 8 SORTIES

Programme TRIAC1.PAS

Ce programme permet de faire clignoter une ampoule à vitesse ajustable par les touches + et - du clavier de l'ordinateur.

Programme SIMULE.PAS

Ce programme permet d'allumer et d'éteindre une lampe aux heures définies par l'utilisateur (10 séquences d'allumage et d'extinction sont possibles).

CARTE MINI TESTEUR PIA/PC / CARTE PIA/PC

Programme BASPIA.PAS

Ce programme permet après envoi d'une valeur sur le port de sortie du PIA d'effectuer une lecture des entrées.

CARTE 4 ENTRÉES/4 SORTIES / CARTE PIA/PC

Programme ENTSORT.PAS

Ce programme permet de commander les relais de la carte 4 entrées/4 sorties et d'effectuer une lecture des entrées.

CARTE CLAVIER NUMÉRIQUE / CARTE PIA/PC

Programme PCLAV.PAS

Ceprogramme affiche la valeur de la touche actionnée sur le mini clavier.

Programme PCLAV2.PAS

Ce programme permet d'entrer un code secret au clavier numérique de l'ordinateur. La composition de ce code et l'appui sur la touche rouge, qui fait office de touche Enter (entrée) sur le mini clavier, permet la comparaison des deux codes et l'affichage d'un message code correct ou non correct suivant le cas.

CARTE CHENILLARD PC / CARTE PIA/PC

Programme CHENIL1.PAS

Ce programme permet le défilement des 8 lampes de la carte chenillard.

CARTE VISUALISATION DES BATTEMENTS CARDIAQUES / CARTE 4E/4S / CARTE PIA/PC

Programme CŒUR1.PAS

Ce programme permet de visualiser par un bip de l'ordinateur ainsi qu'un affichage écran le rythme cardiaque.

CARTE BLOC 2 AFFICHEURS / CARTE PIA/PC

Programme AFF1.PAS

Ce programme allume le premier afficheur avec le nombre 3.

Programme AFF2.PAS

Ce programme allume les 2 afficheurs avec le nombre 13.

Programme AFF3.PAS

Ce programme permet le comptage ou le décomptage des secondes par appui sur l'un ou l'autre des deux poussoirs du bloc 2 afficheurs.

DIGIVOLT / CARTE PIA/PC

Programme DIGI1.PAS

Ce programme permet l'affichage à l'écran de l'ordinateur de la valeur de la tension relevée par Digivolt.

Programme DIGI2.PAS

Ce programme affiche la valeur de la tension relevée par Digivolt en tenant compte de la gamme sélectionnée.

CARTE MOTEUR PAS À PAS / CARTEPIA/PC

Programme MOTEUR1.PAS

Ce programme permet la commande en Pas complet du moteur Pas à Pas dans le sens inverse des aiguilles d'une montre.

Programme MOTEUR2.PAS

Ce programme, comparable au précédent, utilise des instruction différentes pour commander le moteur Pas à Pas en Pas complet dans le sens horaire.

Programme MOTEUR3.PAS

Ce programme permet de faire tourner le moteur du nombre de demi Pas dessinés dans le sens horaire.

Programme MOTEUR4.PAS

Ce programme permet de sélectionner le sens de rotation du moteur et de faire varier sa vitesse de rotation par les touches + et - du clavier de l'ordinateur.

CARTE PROGRAMMATEUR D'EPROM / CARTE PIA/PC

Programme REG-PGM.PAS

Ce programme permet à l'aide d'un oscilloscope le réglage de l'ajustage PGM.

Programme ROMVIERGE.PAS

Ce programme permet de tester la virginité d'une Reprom.

Programme ROMCOPIE.PAS

Ce programme permet de dupliquer le contenu d'une Reprom dans une autre Reprom du même type.

Programme ROMVERIF.PAS

Ce programme permet de comparer le contenu de 2 Reprom et d'indiquer à l'utilisateur si elles sont identiques ou pas.

CARTE 4 ENTRÉES/4 SORTIES / CARTE PIA/PC

Programme ENTREE.PAS

Ce programme teste les 4 entrées de la carte 4 entrées/4 sorties et indique si celles-ci sont au repos ou activées.

Programme RESERVE.PAS

Ce programme gère le remplissage automatique d'un réservoir lorsque le niveau d'eau contenu dans celui-ci est minimal.

BARRIERE ET TÉLÉCOMMANDE INFRAROUGE / CARTE 4E/4S / CARTE PIA/PC

Programme INFRABAR.PAS

Ce programme permet le comptage de personnes ou d'objets ayant coupé le faisceau.

Programme INFRATEL.PAS

À chaque impulsion du bouton poussoir de l'emetteur infrarouge, le programme permet d'alimenter et de couper l'alimentation du relais n 1 de la carte 4E/4S

Sommaire

Les montages de *PC et Robotique* sont disponibles soit en kit, soit en carte montée et testée, chez les revendeurs agréés*.

Une liste des points de vente, ainsi que la tarification des montages, vous seront envoyées à votre demande, contre enveloppe timbrée à votre adresse.

Envoyez votre demande à :

ELECTROME
Service PC et Robotique
Zone industrielle Bordeaux Nord
Cedex 23
33083 BORDEAUX Cedex

* Si vous êtes revendeur, devenez point de vente agréé des kits PC et Robotique.

Les conditions : envoyez votre demande sur papier à en tête de votre société à l'adresse ci-dessus *« Service Revendeur Agréé »*.

Introduction

L'ordinateur a envahi notre environnement. Sur les lieux de notre travail, il fait parti du décor et s'avère indispensable. À la maison, il trône souvent sur la table de la salle à manger ou bien il monopolise notre bureau.

Adapté pour résoudre de multiples problèmes, on se rend compte rapidement qu'après avoir épuisé son côté ludique, gérer le budget familial et taper quelques réponses à des courriers, on ne sait plus très bien à quoi l'occuper.

Et pourtant, on devine que, dans son boîtier gorgé d'électronique, il détient un potentiel énorme que l'on n'utilise pas.

L'INTERFACE

L'interface est la porte que l'on ouvre vers l'extérieur.

Grâce à l'interface, l'ordinateur pourra saisir des informations venant de tous types de capteurs : température, humidité, lumière, son, pression, infrarouge, contact d'interrupteur ou d'ILS, de position, tension, courant etc. et après traitement de ces informations, commander un actionneur approprié : allumage lumière, sirène, moteur, relais, électrovanne, afficheur, etc.

ORGANISATION DE L'INTERFACE

L'ordinateur la considère comme des positions de sa mémoire. C'est-à-dire que lorsqu'il sélectionne ces positions, il peut y écrire ou y lire des données.

En écriture, ces données, grâce à l'interface, seront transmises de la sortie de celui-ci vers l'extérieur.

En lecture, les données extérieures, présentes à l'entrée de l'interface, seront transmises à l'ordinateur.

La carte huit sorties

Pour l'ordinateur, elle correspond à l'adresse &H310 de sa mémoire. Il peut seulement y écrire des données qui seront transmises vers sa sortie.

Par l'instruction basic OUT suivie de l'adresse vers laquelle on veut écrire, et de l'information à y écrire, on transmettra ainsi des données vers la sortie de l'interface.

Une instruction OUT &H310,255 transmet à l'interface un octet dont les huit bits sont tous à 1.

Une instruction OUT &H310, 1 transmet à l'interface un octet dont seul le premier bit est à 1.

OUT &H310, 129 le premier et le huitième sont à 1

OUT &H310, 0 tous les bits sont à zéro.

La carte PIA/PC

Avec elle on entre de plein pied dans l'interfaçage. Elle offre la possibilité de travailler sur trois octets, chacun pouvant être configuré en sortie ou en entrée.

On disposera donc de trois adresses permettant de lire ou d'écrire ces octets plus une adresse propre à l'interface permettant de configurer celui-ci suivant nos besoins.

L'instruction OUT transmettra nos commandes vers l'interface. L'instruction INP permettra de lire les entrées de l'interface en les transmettant à l'ordinateur.

① Carte 8 sorties pour PC

Une interface simple pour PC permettant de commander huit sorties. Cette interface réalisée à partir de composants courants est programmable en BASIC, Pascal ou assembleur etc.

Elle permet de commander huit sorties pouvant délivrer chacune un courant de 500 mA. Insérée directement à l'intérieur du micro (dans un des connecteurs d'extension) elle est directement alimentée par le + 5 V de celui-ci.

PRINCIPE

Un circuit mémoire 8 bits (74 LS 377) est chargé à partir du Bus de donnée du PC lorsqu'une adresse mémoire bien précise de celui-ci est sélectionnée (*fig.* 1.1).

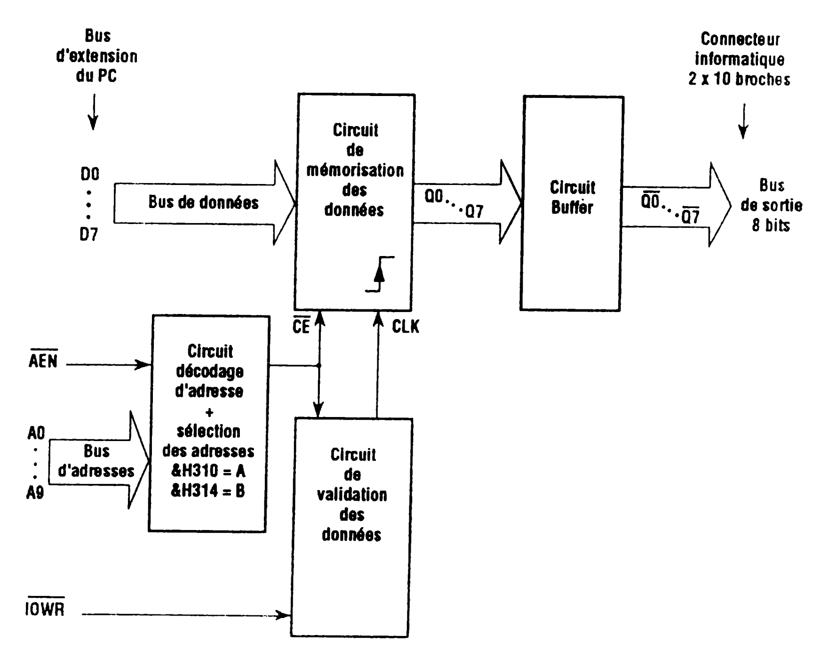


Fig. 1.1

TECHNOLOGIE

L'interface 8 sorties/PC est constituée autour d'un circuit intégré 74 LS 377 permettant de mémoriser les données, associé à un circuit de décodage d'adresse, ainsi qu'à un circuit buffer.

L'interface permet de commander huit sorties différentes.

COMPATIBILITÉ

La carte 8 sorties/PC peut être connectée à tous les ordinateurs compatibles PC : PC/XT, PC/AT.

L'interface est à enficher dans l'un des connecteurs libres (8 bits) de votre ordinateur.

Il y a possibilité également de mettre deux cartes dans le même ordinateur grâce à un jeux de strap sélectionnant deux adresses différentes.

CIRCUIT DÉCODAGE D'ADRESSE

La zone mémoire réservée pour les cartes connectables de votre PC se situe à l'adresse 300 à 31 F (hexadécimal)..

Seules les adresses hexadécimales &H310 et &H314 sont utilisées. Le branchement d'un strap permet de sélectionner la carte huit sorties à l'une ou l'autre de ces adresses (strap en A : adresse = &H310, strap en B : adresse = &H314).

Quand l'adresse présente sur le bus de l'ordinateur est correcte, la sortie (Pin 9) du circuit intégré 74 LS 133 passe au niveau bas (0 V), validant ainsi le circuit de mémorisation (74 LS 377).

Hexadécimal	Strap en A	Strap en B
Adresse de la carte 8 sorties	&H310	&H314

	Bits du Bus d'adresse	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
STRAP (A)	Code binaire	1	1	0	0	0	1	0	0	0	0
	Code hexadécimal	3		1				0			
STRAP (B)	Code binaire	1	1	0	0	0	1	0	1	0	0
	Code hexadécimal	3		1				4			

CIRCUIT DE MÉMORISATION DES DONNÉES

La mémorisation des données est réalisée par une octuple bascule de type Data (74 LS 377).

Si la carte n'est pas adressée, CE sera au niveau haut (+ 5 V). Les sorties Q0 à Q7 ne changent pas d'état (mémorisation des Données). Si l'adresse de la carte est sélectionnée, CE passera au niveau bas et les Données (D0 ... D7) présentent à l'entrée des bascules seront transférées sur les sorties (Q0 ... Q7) lors de la phase écriture de l'ordinateur (IOWR = niveau bas).

CE	CLK	Bus de sortie
H	X	pas de changement d'état
L	Φ	$D_n = Q_n$ transfert des Données

H : niveau haut

X : niveau indifférent

Φ : actif au front montant

L : niveau bas

CIRCUIT BUFFER (*fig.* 1.2)

Constitué par le circuit intégré ULN 2803 A, il permet d'amplifier le signal de sortie. Ce circuit contient un réseau de huit transistors Darlingtons

supportant un VCE max de 50 V et un courant IC maximal de 500 mA. Ce circuit vous permet d'alimenter directement un actionneur (relais, petit moteur, ..., voyant) grâce à une alimentation continue externe.

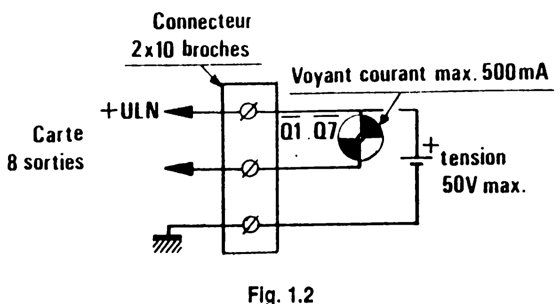


Fig. 1.2

SCHÉMA DE PRINCIPE (*fig.* 1.3)

Les sorties $\overline{Q0}$ et $\overline{Q1}$ sont reliées à la prise 2 × 10 broches de la carte et, en même temps, aux deux borniers (plots moins des borniers), ce qui permet pour des expérimentations simples (voir commande de LED, buzzer, triac, relais) de connecter ces petits montages à la carte, fils à fils. Les plots positifs sont reliés au plus à travers des résistances de 470 Ω, ce qui limite le courant entre les points plus et moins à 10 mA.

MONTAGE

Souder dans l'ordre suivant et le plus près possible du circuit :

- les résistances,
- les condensateurs (attention à la polarité du tantale),
- le réseau repéré par un point,
- les circuits intégrés repérés par un point ou une encoche.

Ne pas oublier les soudures des pattes côté composants : toutes les pastilles du circuit doivent recevoir de la soudure.

Souder ensuite les deux diodes repérées par une bague, les borniers à vis (en orientant les sorties fil vers l'extérieur du circuit) et le connecteur 2 × 10 broches.

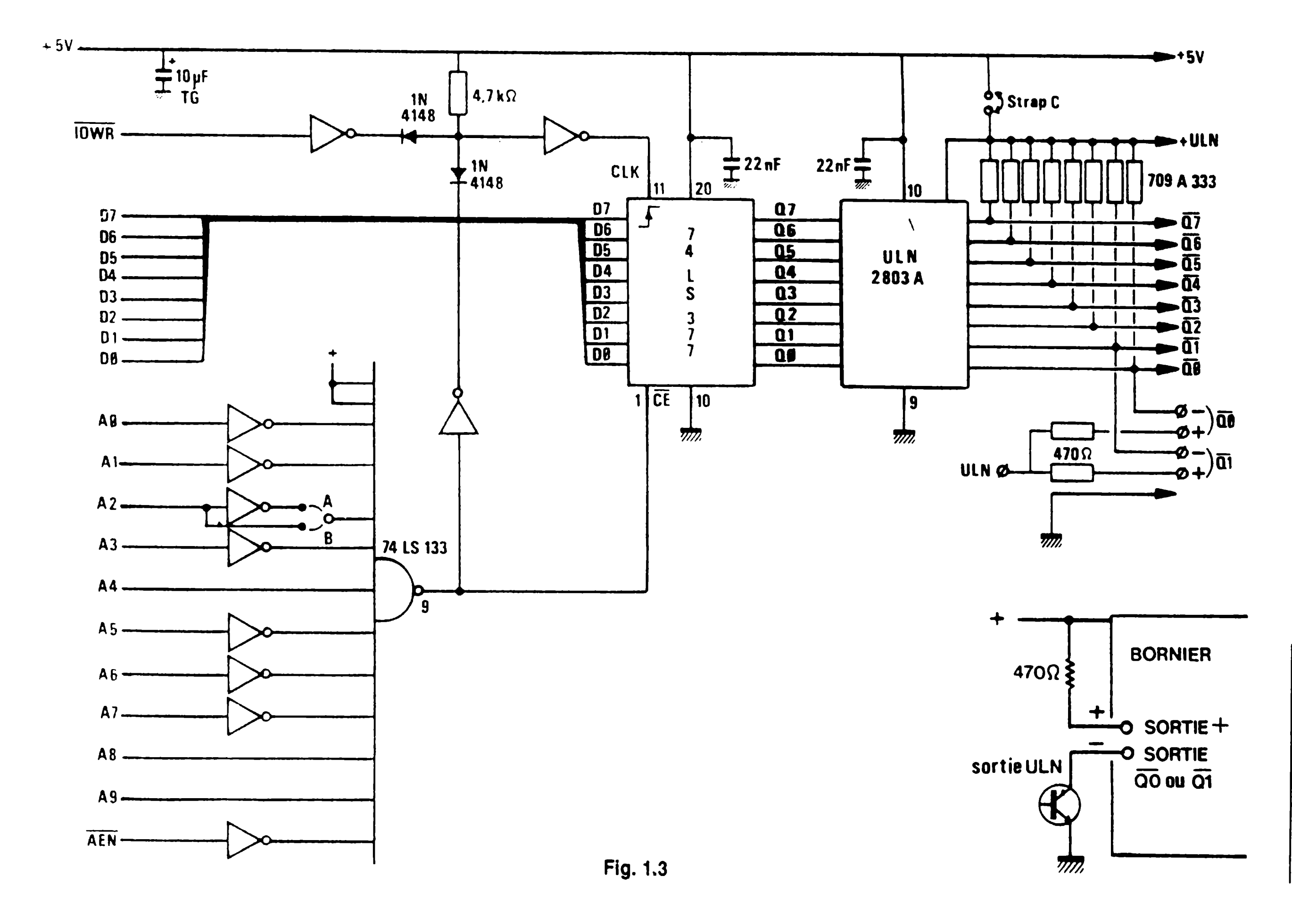

+5V
+5V
10µF TG
IOWR
1N 4148
4,7 kΩ
1N 4148
CLK
11
20
22 nF
22nF
10
Strap C
+ULN
709 A 333
D7 D6 D5 D4 D3 D2 D1 DØ
74 LS 377
Q7 Q6 Q5 Q4 Q3 Q2 Q1 QØ
ULN 2803 A
1 CE
10
9
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9
AEN
A
B
74 LS 133
9
470Ω
ULN Ø
BORNIER
470Ω
SORTIE +
SORTIE
QO ou Q1
sortie ULN

Fig. 1.3

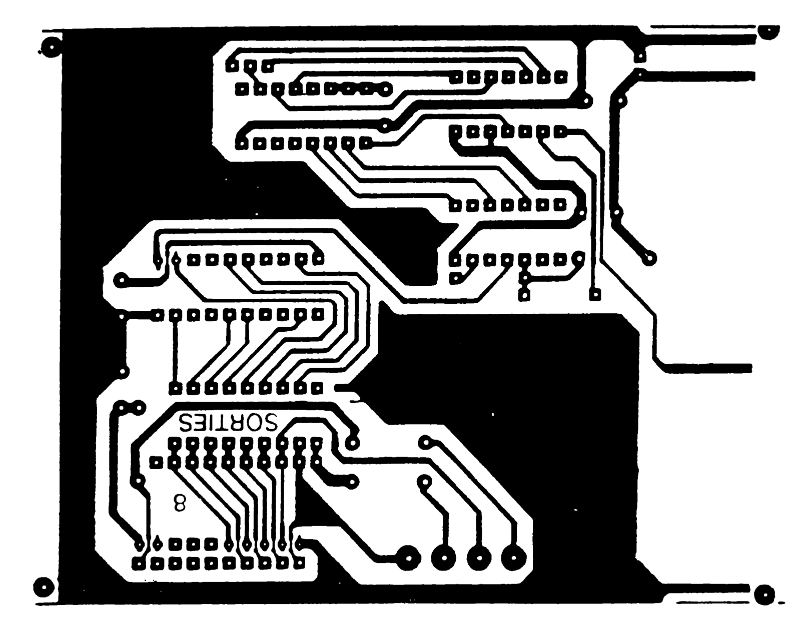


Fig. 1.4

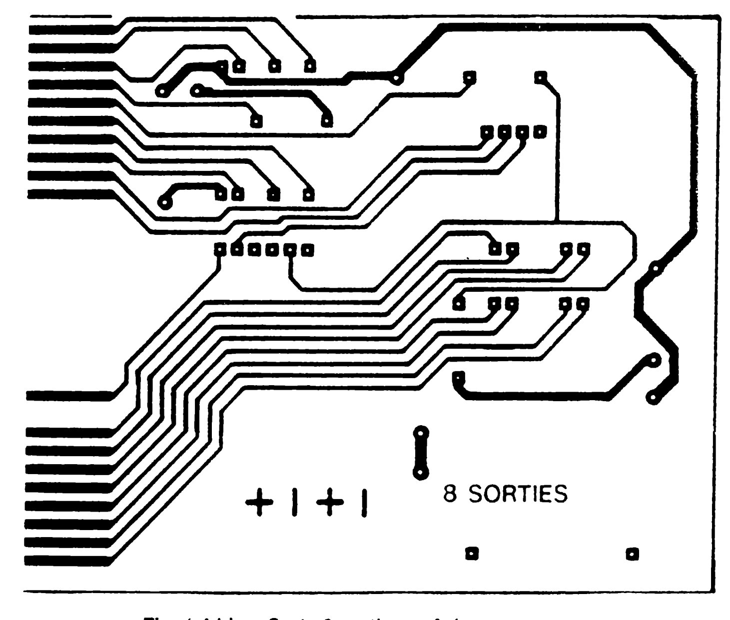


Fig. 1.4 bis – Carte 8 sorties - côté composants

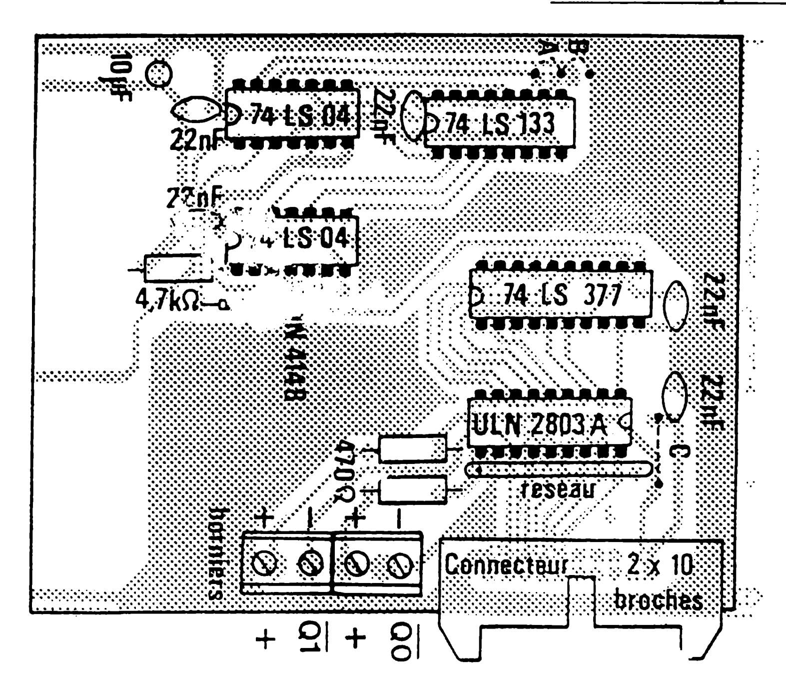


Fig. 1.5

LISTE DES COMPOSANTS

Résistances :
2 x 470 Ω (jaune violet marron);
1 x 4,7 kΩ (jaune violet rouge);
1 x réseau 709 A 333 ou 710 A 333.

Condensateurs :
5 x 22 nF à 100 nF;
1 x 10 µF/10 V tantale goutte.

Semi-conducteurs :
2 x 1N 4148;
2 x 74 LS 04;
1 x 74 LS 133;
1 x 74 LS 377;
1 x ULN 2803.

Divers
2 borniers 2 plots pour CI
1 connecteur mâle coudé 2 x 10 broches.

Souder le strap en A ou en B selon les adresses que vous avez choisies.

strap en B adresse : &H314

strap en A adresse : &H310

Strap en C :

— dans le cas d'une utilisation avec le connecteur 2 × 10 broches, ne pas mettre le strap,

— dans le cas d'une utilisation avec sortie sur borniers, mettre le strap.

On dispose de deux sorties sur borniers ($\overline{Q0}$ et $\overline{Q1}$) pouvant fournir un courant de 10 mA permettant de connecter directement deux LED de visualisation, la carte relais ou la carte triac (*fig.* 1.4).

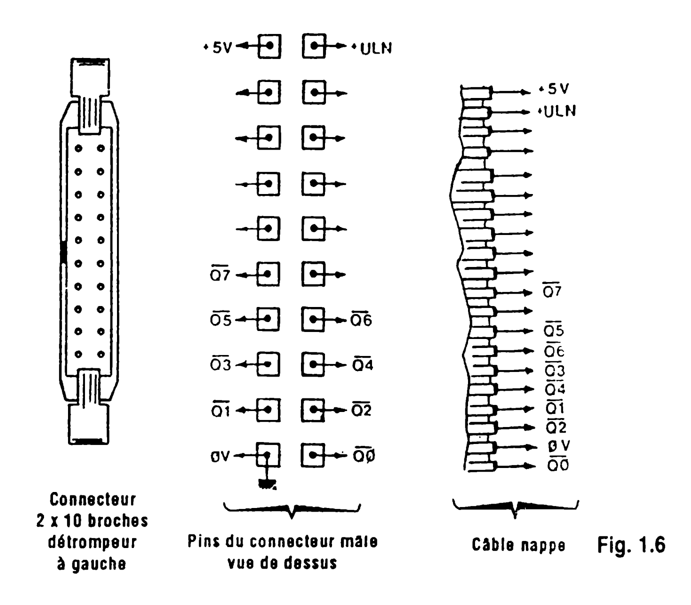


Fig. 1.6

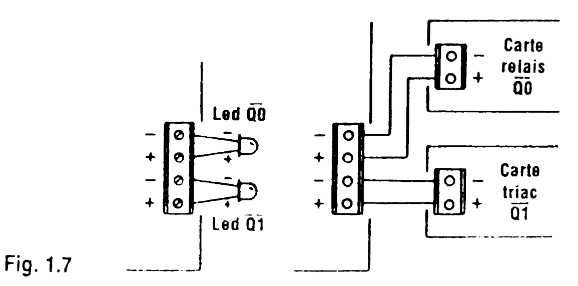


Fig. 1.7

② Allumage de diodes électroluminescentes

À travers les sorties de la carte 8 sorties, nous pouvons connecter directement une diode électroluminescente. En effet, une résistance de 470 Ω en série avec le plus limite le courant à environ 10 mA et protège chacune des sorties contre un court-circuit.

Sous GW BASIC :

En composant OUT &H310,3 puis entrée (*enter*), les deux diodes s'allument.

Les sorties $\overline{Q0}$ et $\overline{Q1}$ sont passées à 0 V et les diodes s'allument.

En composant OUT &H310,0 puis entrée (*enter*), les deux diodes s'éteignent.

$\overline{Q0}$ et $\overline{Q1}$ sont repassées à plus.

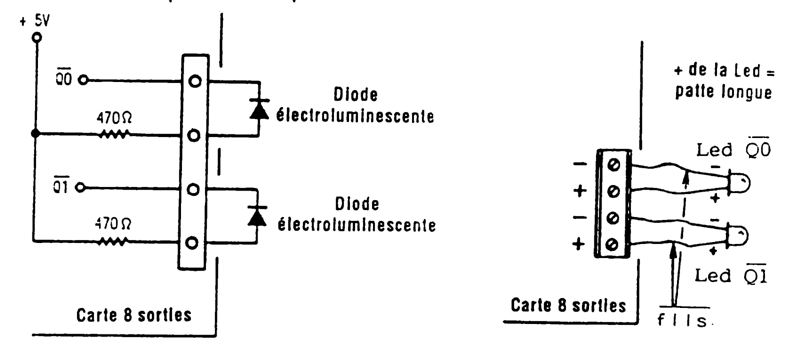


Fig. 2.1

PROGRAMME LED1

Programme de clignotement des deux leds.

```
10  ------LED1----------
20 CLS
30 INPUT" Nombre de boucles de temporisation:";A
40 OUT &H310,3
50 FOR I=1 TO A:NEXT I
60 A$=INKEY$
70 IF A$=" " THEN GOTO 20
80 OUT &H310,0
90 FOR I=1 TO A:NEXT I
100 GOTO 40
```

en remplaçant OUT &H310,3 par OUT &H310,1 seule la Led du haut clignote.

EXPLICATION DU PROGRAMME LED1

ligne 20 : RAZ écran,

ligne 30 : entrée dans A le nombre de cycle qui définira la vitesse du clignotement,

ligne 40 : sortie adresse 310 code 3 (allumage des LED),

ligne 50 : A fois la boucle FOR NEXT I (tempo LED allumées),

ligne 60 : entrée dans A$ sans arrêter le programme,

ligne 70 : si A$=" " (barre espace) aller à la ligne 20 (pour recomposer une vitesse de défilement),

ligne 80 : sortie adresse 310 code 0 (extinction des LED),

ligne 90 : A fois la boucle FOR NEXT I (tempo LED éteintes),

ligne 100 : retour à la ligne 40 (recommence le programme).

LANCEMENT DU PROGRAMME

Faire RUN"LED1" puis entrée (*enter*).

Le programme demande le nombre de boucles de temporisation.

Composer 500, puis entrée (*enter*).

Les LED doivent clignoter. Appuyer sur barre espace, le programme redemande le nombre de boucles.

Composer un nombre plus grand ou plus petit que 500 pour ralentir ou accélérer le clignotement.

Pour sortir du programme, appuyer sur CTRL et Pause (Breack).

③ De la musique et des boucles

À l'aide d'un transducteur piézo (*fig.* 3.1), nous allons mettre en évidence la vitesse de calcul de l'ordinateur.

Sur la carte huit sorties, mettre deux fils séparés dans le domino du haut, puis remettre la carte dans l'ordinateur. Réunir les fils à un petit buzzer (transducteur piézo), il n'y a pas de sens à respecter.

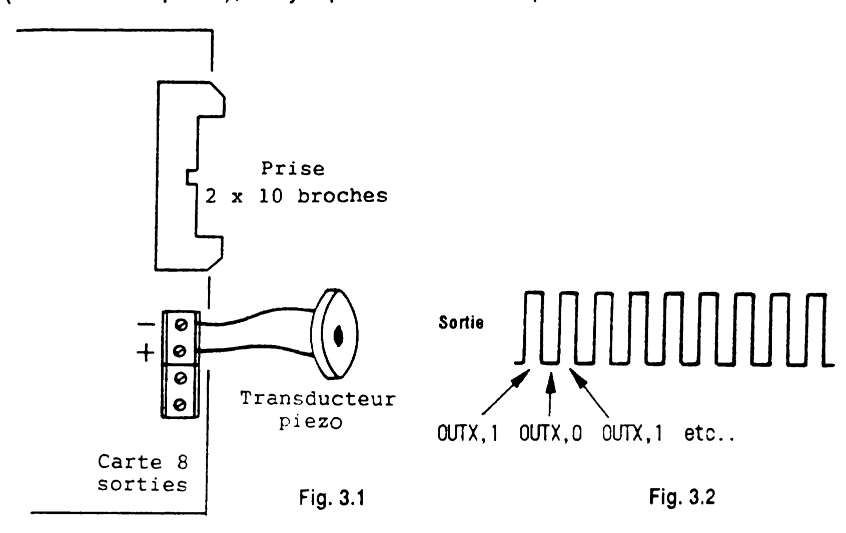


Fig. 3.1

Fig. 3.2

Allumer l'ordinateur, charger le Basic.

Le Basic étant chargé, taper :

```
10 OUT &H310, 1
20 OUT &H310, 0
30 GOTO 10
```

Faire RUN.

Le buzzer doit émettre un son continu plus ou moins aigu suivant la vitesse de l'horloge de l'ordinateur.

En effet :

la ligne 10 du programme met à 0 la sortie du – bornier,

la ligne 20 la met au plus,

la ligne 30 reboucle le programme en 10,

et ainsi de suite

La sortie moins du bornier de la carte varie donc entre 0 et + suivant une fréquence qui dépend du temps d'horloge du micro-ordinateur (*fig.* 3.2).

Arrêter le programme en appuyant sur CTRL et pause (ou *Breack*).

UN BIP-BIP AVEC UNE BOUCLE *FOR... NEXT*

Ajouter au programme :

```
1  INPUT « Durée Bip »; A
5  For X=1 TO A
30 Next X
40 For X=1 TO A
50 Next X
60 GOTO 5
```

Faire RUN « durée Bip »? Composer 200, un Bip-Bip plus ou moins rapide doit retentir.

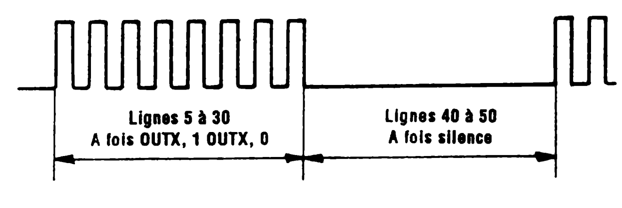


Fig. 3.3

Faire CTRL et Pause pour arrêter le programme. Refaire RUN et composer une valeur plus grande ou plus petite que 200 si vous voulez diminuer ou augmenter la vitesse du Bip-Bip (*fig.* 3.3).

EXPLICATION DU PROGRAMME

ligne 1 : composition dans A du nombre de boucles à faire qui détermineront la durée du Bip, puis celle du silence,

ligne 5 : réalisation de la boucle de 1 à A fois,

lignes 10 à 20 : sortie d'un Bip,

ligne 30 : retour à 5 si le nombre de fois A n'est pas atteint. Si le nombre de fois A est atteint, le programme continue vers 40,

ligne 40 : réalisation de la boucle de 1 à A fois,

ligne 50 : retour à 40 si le nombre de fois n'est pas atteint sinon le programme continue vers 60,

ligne 60 : retour à 5.

④ Carte un afficheur

Cette carte permet d'afficher toutes les combinaisons possibles avec les sept segments et le point décimal d'un afficheur. L'alimentation de l'afficheur est fournie par l'ordinateur.

SCHÉMA (*fig.* 4.1)

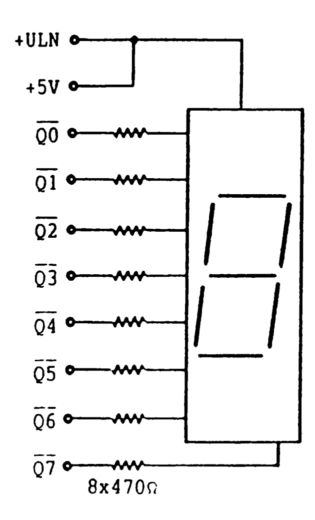


Fig. 4.1

PRINCIPE

L'afficheur étant à anodes communes, lorsque $\overline{QX}$ passe au niveau logique bas (0 V), le segment concerné s'allume.

La valeur de chacun des segments est indiquée sur la figure 4.2.

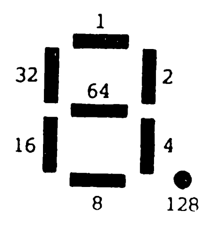


Fig. 4.2

C'est-à-dire qu'en envoyant la valeur 64 à la carte afficheur seul le segment du centre sera allumé. En envoyant 128 on allumera le point. En envoyant 79, on sélectionne les segments 1, 2, 4, 8, 64 soit l'affichage du chiffre 3.

En envoyant 255 (la somme de 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128) les sept segments et le point s'allument.

IMPLANTATION

Attention! Lors du montage de l'afficheur, placer le point décimal en bas à droite.

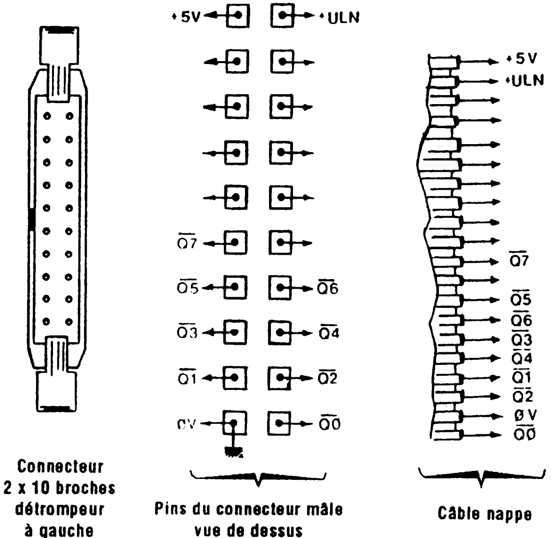


Fig. 4.3

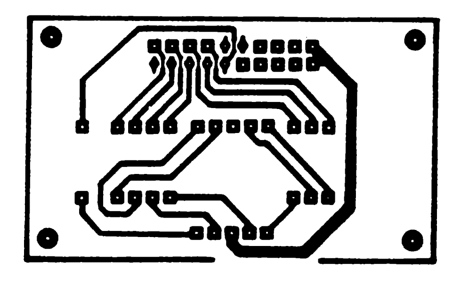


Fig. 4.4

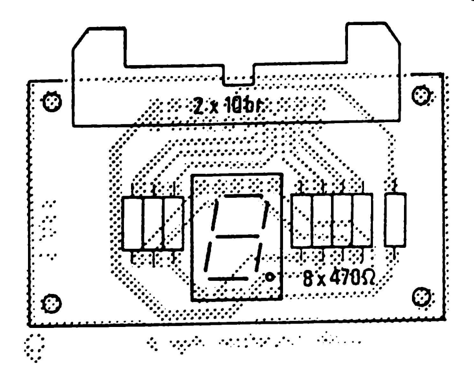
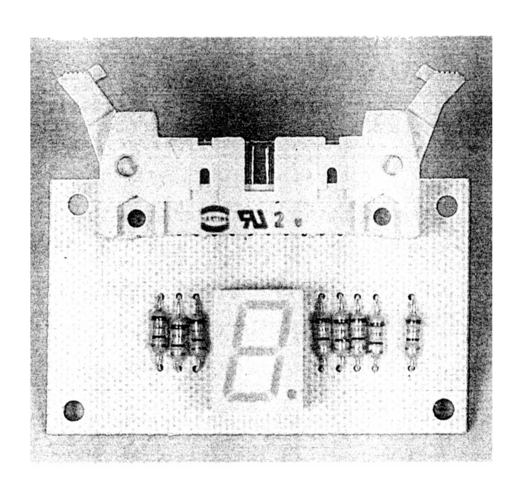


Fig. 4.5

LISTE DES COMPOSANTS

8 × 470 Ω (jaune violet marron);
1 afficheur 13 mm anodes communes;
1 connecteur mâle coudé 2 × 10 broches.



PROGRAMME 7SEGM1

Réunir un câble plat vingt conducteurs (réf. CA20) la carte afficheur à la carte huit sorties.

Sous GW BASIC (Microsoft) :
Taper RUN"7SEGM1"

Ce programme permet d'allumer le segment dont on a composé la valeur.

Après l'affichage sur l'écran de : « Valeur des segments à afficher »

Composer 1 : le segment du haut s'allume,
Composer 0 : l'afficheur est éteint,
Composer 255 : on affiche 8 et le point : tous les segments et le point sont allumés,
Composer 79 : on affiche 3,
Composer 119 : on affiche A,
Composer 121 : on affiche E,
Composer 57 : on affiche C,
Composer 185 : 185 = 57 + 128 on affiche toujours C mais suivi du point.
On pourra ainsi vérifier que :

1=6; 2=91; 3=79; 4=102; 5=109;
6=125; 7=7; 8=127; 9=111; 0=63;

et en ajoutant 128 à chacune de ces valeurs on allume le point avec le chiffre.

Pour sortir du programme composer −1

```
5 '----programme---7SEGM1-------
10 CLS
20 LOCATE 1,25:PRINT" Retour au BASIC ,composer -1 "
30 LOCATE 10,20:PRINT"valeur des segments à afficher:        ";
40 LOCATE,52:INPUT A
50 IF A=-1 THEN CLS:OUT &H310,0:END
60 IF A<0 OR A>255 THEN BEEP:GOTO 30
70 OUT &H310,A
80 GOTO 30
```

EXPLICATION DU PROGRAMME

ligne 10 : RAZ écran,

lignes 20 et 30 : texte à l'écran,

ligne 40 : entrée de la valeur des segments dans A,

ligne 50 : test si A=−1, extinction de l'afficheur (OUT&H310,0 extinction de tous les segments vers sortie) et arrêt du programme,

ligne 60 : test si A<0 ou A>255 retour à composition de la valeur des segments (erreur à la composition),

ligne 70 : sortie à l'adresse &H310 de la valeur de A (commande de l'afficheur),

ligne 80 : retour à la composition de la valeur des segments.

PROGRAMME 7SEGM2

Taper RUN"7SEGM2"

Ce programme affiche le chiffre composé au clavier (de 0 à 9) au-dessus, un bip sonore vous indique une erreur de composition.

```
10 '----programme---7SEGM2-------
20 CLS
30 LOCATE 1,25:PRINT" Retour au BASIC ,composer -1 "
40 LOCATE 10,20:PRINT"Chiffre à afficher:        ";
50 LOCATE,39:INPUT A
60 IF A=-1 THEN CLS:OUT &H310,0:END
70 IF A<0 OR A>9 THEN BEEP:GOTO 40
80 ON A GOSUB 120,130,140,150,160,170,180,190,200
90 GOTO 210
100 OUT &H310,A
110 GOTO 40
120 A=6:GOTO 100
130 A=91:GOTO 100
140 A=79:GOTO 100
150 A=102:GOTO 100
160 A=109:GOTO 100
170 A=125:GOTO 100
180 A=7:GOTO 100
190 A=127:GOTO 100
200 A=111:GOTO 100
210 A=63:GOTO 100
```

EXPLICATION DU PROGRAMME

lignes 20 à 40 : affichage écran,

ligne 50 : entrée du chiffre à afficher dans A,

ligne 60 : test si A=–1, RAZ écran, extinction de l'afficheur et retour au programme,

ligne 70 : test si A<0 ou A>9 bip sonore puis retour ligne 40,

ligne 80 : en fonction de A, saut à 120 si A=1, 130 si A=2, ... 200 si A=9,

ligne 90 : si A=0 saut à 210,

ligne 100 : sortie vers afficheur de la valeur A chargée entre ligne 120 à 210 avec le code afficheur correspondant,

ligne 110 : retour à chiffre à composer,

lignes 120 à 210 : suivant le chiffre composé au clavier, chargement dans A du code afficheur correspondant et retour à 100 si chiffre composé =5, A=5 ligne 80 saut à 160 chargement de la valeur 109 dans A correspondant à 1+32+64+4+8=109 (voir sur figure des valeurs de chaque segment) retour à 100 et sortie de la valeur 109 dans l'afficheur soit affichage de 5

Retour au BASIC, composer 1

PROGRAMME 7SEGM3

Taper RUN "7SEGM3"

Ce programme affiche les dix chiffres suivis du point en permanence. La vitesse de défilement étant accélérée en appuyant sur + et ralentie en appuyant sur –. En appuyant sur la barre d'espacement, on revient au Basic.

```
2 '-------programme----7SEGM3--------
5 CLS:KEY OFF
7, LOCATE 10,17:PRINT" Retour au BASIC, appuyer sur barre espace "
10 LOCATE 12,10
12 PRINT" Défilement plus rapide ,appuyer sur +, moins rapide sur -"
20 TOT=100
30 RESTORE 160
40 FOR I=1 TO 11
50 READ A
60 OUT &H310,A
70  FOR J=1 TO TOT
80 A$=INKEY$
90 IF A$="-" THEN TOT =TOT+5
100 IF TOT<10 THEN GOTO 120
110 IF A$="+" THEN TOT =TOT-5
120 IF A$=" " THEN CLS:OUT &H310,0:END
130 NEXT J
140 NEXT I
150 GOTO 30
160 DATA 6,91,79,102,109,125,7,127,111,63,128
```

EXPLICATION DU PROGRAMME

ligne 20 : constante de temporisation affichage TOT=100,

ligne 30 : initialisation des constantes de la ligne 160,

ligne 40 : pour I=1 à 11,

ligne 50 : lecture constante de la ligne 160,

ligne 60 : vers sortie à l'adresse &H310 (affichage de segments chargés en 50),

ligne 70 : temporisation de 1 à TOT,

lignes 80 à 120 : si entrée dans A$=–, augmentation de la constante de temporisation TOT de 5 si = + et que TOT ne soit pas <20, diminution de TOT de 5 si =espace (" ") retour au Basic en éteignant l'afficheur,

ligne 130 : incrémentation de la boucle de temporisation si =TOT suite vers 140,

ligne 140 : incrémentation I (numéro de la valeur à afficher),

ligne 150 : retour à 30 pour recommencer un cycle d'affichage de 10 chiffres puis du point,

ligne 160 : onze valeurs correspondants à l'affichage de 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, •

⑤ Carte canil

Cette petite carte est composée de huit LED avec chacune sa résistance de limitation en courant. Elle permet de visualiser directement la sortie de la carte huit sorties ou le port A de la carte PIA/PC.

De plus, un bouton poussoir, dans le cas d'utilisation avec la carte PIA/PC, permet de transmettre une donnée à l'ordinateur.

PRINCIPE

Les LED s'allument lorsqu'on trouve un niveau bas sur les ports PA. Le port PB0 est actif au niveau logique haut obtenu par le contact travail du bouton poussoir.

SCHÉMA

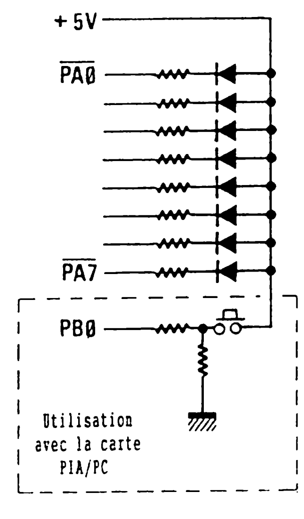


Fig. 5.1

CIRCUIT IMPRIMÉ

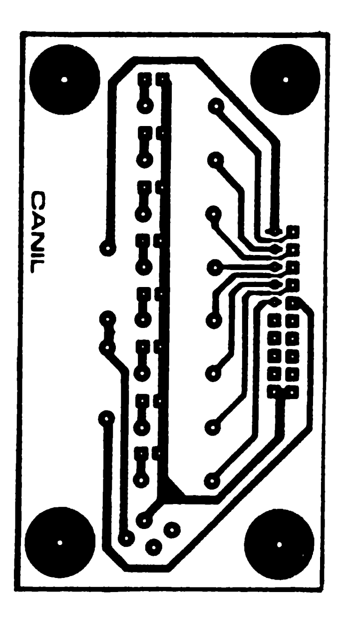


Fig. 5.2

IMPLANTATION

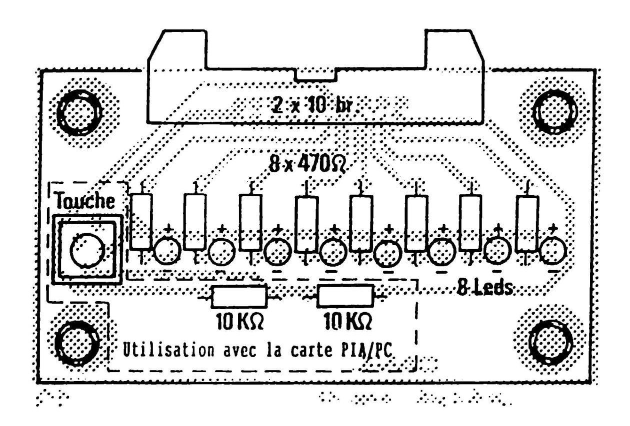


Fig. 5.3

MONTAGE

Souder dans l'ordre et le plus près possible du circuit :

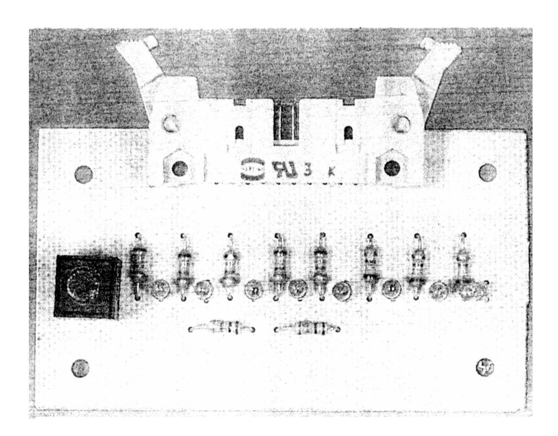
— les dix résistances,

— les LED en respectant la polarité (patte longue au +),

— la touche,

— le connecteur 2 × 10 broches,

— les quatre rivets.



LISTE DES COMPOSANTS

8 × 470 Ω (jaune violet marron);
8 LED rouge 3 mm;
1 connecteur mâle coudé 2 × 10 broches;
4 rivets 4 × 10;
Option pour utilisation avec carte PIA/PC :
2 × 10 kΩ (marron noir orange);
1 touche pour CI.

PROGRAMME 1 : CANIL1

Ce programme permet de vérifier, grâce à l'allumage des Leds, les valeurs transmises par l'interface.

Après avoir connecté la petite carte au câble venant de l'ordinateur, sous GW BASIC, faire RUN"CANIL1".

Le programme vous demande de composer un nombre. Si ce nombre est égal à –1, retour au Basic.

Composer 1, puis entrée (enter), la première Led s'allume. Composer 2, la deuxième s'allume. 3, les deux premières. 4 la troisième. 255, toutes les Leds. 0, toutes les Leds sont éteintes. 256, l'ordinateur émet un Bip pour vous signaler que vous dépassez la valeur maximum. Vérifier le tableau ci-dessous :

Led n	1	2	3	4	5	6	7	8
Valeur binaire	1	2	4	8	16	32	64	128

```
10 '--programme---CANIL1---
20 CLS
30 KEY OFF
40 COLOR 0,7:LOCATE 5,22:PRINT " Affichages sorties carte PIA ":COLOR 7,0
45 LOCATE 7,20:PRINT" Pour revenir au BASIC composer -1 "
50 'config pia-------------
60 P=&H313
70 O=&H310
80 R=&H311
90 OUT P,&H82
100 LOCATE 12,20:PRINT " Votre choix:          ";:LOCATE ,40:PRINT A;
110 LOCATE ,34:INPUT A
115 IF A=-1 THEN CLS:END
120 IF A>255 OR A<0 THEN BEEP:GOTO 100
130 OUT O,A
140 GOTO 100
```

EXPLICATION DU PROGRAMME CANIL1

ligne 20 : RAZ écran,

ligne 30 : extinction vingt-cinquième ligne,

lignes 40 et 50 : affichage des directives,

lignes 60 à 90 : configuration du PIA 0 en sortie, R en entrée,

lignes 100 à 120 : entrée et affichage de la valeur à transmettre au PIA. Si entrée =–1, retour au basic, si elle est supérieure à 255 ou inférieure à 0, il y a émission d'un bip et retour à l'entrée valeurs,

ligne 130 : sortie de la valeur de A vers l'interface,

ligne 140 : retour à composition.

PROGRAMME 2 : CANIL2

Utilisation avec carte PIA/PC

Lorsque la touche est appuyée, elle transfère la valeur 1 dans le port B.

Le programme CANIL2 est identique à CANIL1, mais pour avoir accès à la composition d'un nouveau code, il faut appuyer sur la touche de la carte.

Sous GW BASIC, faire

RUN"CANIL2".

Composer par exemple 2, puis entrée (*enter*). La deuxième LED s'allume. Le message appuyer sur la touche pour nouveaux choix apparaît à l'écran.

Appuyer sur la touche de la carte CANIL, le message disparaît et vous pouvez de nouveau composer un code.

Faire –1 pour retour au BASIC.

```
10  --programme---CANIL2---
20 CLS
30 KEY OFF
40 COLOR 0,7:LOCATE 5,22:PRINT " Affichages sorties carte PIA ":COLOR 7,0
50 LOCATE 7,20:PRINT" Pour revenir au BASIC composer -1 "
60 'config pia-------------
70 P=&H313
80 O=&H310
90 R=&H311
100 OUT P,&H82
110 LOCATE 12,20:PRINT " Votre choix:           ";:LOCATE ,40:PRINT A;
120 LOCATE ,34:INPUT A
130 IF A=-1 THEN CLS:END
140 IF A>255 OR A<0 THEN BEEP:GOTO 110
150 OUT O,A
160 LOCATE 15,15:PRINT " appuyer sur la touche pour nouveau choix ";
170 ENT=(INP(R) AND 1)
180 IF ENT =1 THEN LOCATE ,15:PRINT SPC(50):GOTO 110
190 GOTO 170
```

EXPLICATION DU PROGRAMME

Identique à CANIL1 jusqu'à 150.

ligne 160 : message, appuyer sur touche à l'écran,

ligne 170 : entrer dans ENT la valeur du Port B en ne sélectionnant que le 1 (les autres poids sur Port B étant en l'air peuvent transmettre des valeurs aléatoires),

ligne 180 : si ENT=1 (touche enfoncée), effacement à l'écran de la ligne 160 et retour au choix,

ligne 190 : retour à test si touche enfoncée,

PROGRAMME 3 : CANIL 3

Utilisation avec la carte PIA/PC

Ce programme permet de composer vingt séquences lumineuses s'enchaînant l'une derrière l'autre. Composer

RUN"CANIL3"

```
10 '--programme---CANIL3---
20 CLS
30 KEY OFF
40 COLOR 0,7:LOCATE 5,22:PRINT " Affichages sorties carte PIA ":COLOR 7,0
50 LOCATE 7,17:PRINT" Pour revenir au BASIC appuyer sur touche  -1 "
60 'config pia-------------
70 P=&H313
80 O=&H310
90 R=&H311
100 OUT P,&H82
110 DIM A (20)
120 L=0
130 FOR I=1 TO 20
140 LOCATE ,L*17+1:PRINT " Valeur ";:PRINT USING"## :";I;
150 INPUT;A(I)
160 IF A(I)=-1 THEN CLS:OUT O,0:END
170 IF A(I)<0 OR A(I)>255 THEN BEEP:GOTO 140
180 L=L+1:IF L=4 THEN PRINT" ":L=0
190 NEXT I
200 LOCATE 7,17:PRINT" Pour revenir au BASIC appuyer sur touche    "
210 FOR I=1 TO 20
220 OUT O,A(I)
230 ENT=INP(R) AND 1
240 IF ENT=1 THEN CLS:OUT O,0:END
250 FOR J=1 TO 200:NEXT J
260 NEXT I
270 GOTO 210
```

EXPLICATION DU PROGRAMME

lignes 1 à 100 : identique aux autres programmes CANIL,

ligne 110 : déclaration du fichier A 20 enregistrements,

ligne 120 : comptage édition sur chaque ligne=0,

ligne 130 : boucle de vingt entrées,

ligne 140 : à la position Lx17+1 (L=numéro édition sur la ligne) affiche valeur suivie du numéro de la valeur (de 1 à 20),

ligne 150 : valeur dans A (I). I représente le numéro de la valeur de 1 à 20,

ligne 160 : si entrée =–1, retour au basic,

ligne 170 : si entrée inférieure à 0 ou supérieure à 255, erreur, bip retour à 140,

ligne 180 : numéro édition = numéro édition +1. Si =4, passage à la ligne suivante. Remise numéro édition à 0,

ligne 190 : retour à 130 si I n'est pas à 20,

ligne 200 : affichage retour au Basic, appuyer sur touche (sur la mini carte),

ligne 210 : boucle pour I = 1 à 20,

ligne 220 : sortie vers la mini plaque de la valeur entrée dans A (I représentant le numéro de la valeur de 1 à 20),

ligne 230 : entrée dans ENT de la valeur du poids 1 du port B,

ligne 240 : si valeur entrée = 1 (touche enfoncée) RAZ écran sortie de la valeur 0 sur port A pour éteindre toutes les LED. Retour au basic,

ligne 250 : boucle de temporisation permet à chaque séquence lumineuse de rester allumée un certain temps. Si le temps est trop court, augmenter la valeur 200 (ex. FOR J = 1 TO 250) s'il est trop long, diminuer 200 (ex. FOR J = 1 TO 120),

ligne 260 : on recommence la boucle si I n'est pas à 20,

ligne 270 : retour au début de la boucle pour refaire les 20 séquences lumineuses.

Pour sortir des séquences lumineuses, il faudra appuyer sur la touche de la mini carte.

Les deux programmes CANIL1S et CANIL3S sont destinés à commander la carte CANIL de la page 00 à l'aide de la carte 8 sorties.

Les programmes CANIL1, CANIL2, CANIL3 commandent cette même carte mais à l'aide de la carte PIA/PC 24 entrées/sorties.

EXEMPLE DE SÉQUENCES À COMPOSER

Valeurs	1 = 1	9 = 64	17 = 255
	2 = 2	10 = 32	18 = 0
	3 = 4	11 = 16	19 = 255
	4 = 8	12 = 8	20 = 0
	5 = 16	13 = 4	
	6 = 32	14 = 2	
	7 = 64	15 = 1	
	8 = 128	16 = 0	

La LED allumée se déplacera en aller-retour de la première à la huitième, puis les huit LED clignoteront deux fois avant de recommencer l'aller-retour.

PROGRAMME CANIL1S

Avec la carte huit sorties.

Ce programme permet de vérifier, grâce à l'allumage des LEDs, les valeurs transmises par l'interface.

Après avoir connecté la petite carte au câble venant de l'ordinateur, sous GW BASIC, faire RUN"CANIL1S".

Le programme vous demande de composer un nombre. Si ce nombre est égal à –1, retour au basic.

Composer 1, puis entrée (*enter*), la première LED s'allume. Composer 2, la deuxième LED s'allume. Composer 3, les deux premières. Composer 4, la troisième. Composer 255, toutes les LED. Composer 0, toutes les LED sont éteintes. Composer 256, l'ordinateur émet un bip pour signaler que vous dépassez la valeur maximale. Vérifiez le tableau ci-dessous.

LED n	1	2	3	4	5	6	7	8
Valeur binaire	1	2	4	8	16	32	64	128

```
10 '--programme---CANIL1S--
20 CLS
30 KEY OFF
40 COLOR 0,7:LOCATE 5,22:PRINT " Affichages sorties carte 8 s ":COLOR 7,0
50 LOCATE 7,20:PRINT" Pour revenir au BASIC composer -1 "
60 LOCATE 12,20:PRINT " Votre choix:          ";:LOCATE ,40:PRINT A;
70 LOCATE ,34:INPUT A
80 IF A=-1 THEN CLS:END
90 IF A>255 OR A<0 THEN BEEP:GOTO 60
100 OUT &H310,A
110 GOTO 60
```

EXPLICATION DU PROGRAMME CANIL1S

ligne 20 : RAZ écran,

ligne 30 : extinction de la vingt-cinquième ligne,

lignes 40 et 50 : affichage directive,

lignes 60 à 90 : entrée et affichage de la valeur à transmettre au PIA. Si entrée =–1, retour au basic, si entrée supérieure à 255 ou inférieure à 0, l'ordinateur émet un bip, et retour à l'entrée valeur,

ligne 100 : sortie de la valeur de A vers l'interface,

ligne 110 : retour à composition.

PROGRAMME CANIL3S

Ce programme permet de composer vingt séquences lumineuses s'enchaînant l'une derrière l'autre. Composer RUN"CANIL3S".

```
10 '--programme---CANIL3S--
20 CLS
30 KEY OFF
40 COLOR 0,7:LOCATE 5,22:PRINT " Affichages sorties carte 8 S ":COLOR 7,0
50 LOCATE 8
60 DIM A (20)
70 L=0
80 FOR I=1 TO 20
90 LOCATE ,(L*17)+4:PRINT " Valeur ";:PRINT USING"## :";I;
100 INPUT;A(I)
120 IF A(I)<0 OR A(I)>255 THEN BEEP:GOTO 90
130 L=L+1:IF L=4 THEN PRINT" ":L=0
140 NEXT I
145 LOCATE 6,14:PRINT" Pour revenir au BASIC appuyer sur barre espace "
150 FOR I=1 TO 20
160 OUT &H310,A(I)
170 A$=INKEY$
180 IF A$=" "THEN CLS:OUT &H310,0:END
190 FOR J=1 TO 200:NEXT J
200 NEXT I
210 GOTO 150
```

EXPLICATION DU PROGRAMME

lignes 1 à 40 : identique aux autres programmes CANIL1S,

ligne 50 : adressage ligne 8,

ligne 60 : déclaration du fichier A 20 enregistrements,

ligne 70 : comptage édition sur chaque ligne=0,

ligne 80 : boucle de vingt entrées,

ligne 90 : à la position Lx17+4 (L=numéro édition sur la ligne) affiche valeur suivie du numéro de la valeur (de 1 à 20),

ligne 100 : valeur dans A (I). I représente le numéro de la valeur de 1 à 20,

ligne 120 : si entrée inférieure à 0 ou supérieure à 255, erreur, émission d'un bip, et retour à 90,

ligne 130 : numéro édition = numéro édition +1. Si entrée =4, passage à la ligne suivante. Remise numéro édition à 0,

ligne 140 : retour à 80 si I n'est pas à 20,

ligne 145 : affichage,

ligne 150 : boucle pour I = 1 à 20,

ligne 160 : sortie vers la mini plaque de la valeur entrée dans A (I représentant le numéro de la valeur de 1 à 20),

ligne 170 : si barre espace appuyée,

ligne 180 : RAZ écran, sortie=0, retour au basic,

ligne 190 : boucle de temporisation permet à chaque séquence lumineuse de rester allumée un certain temps. Si le temps est trop court, augmenter la valeur 200 (ex. FOR J = 1 TO 250) si il est trop long, diminuer 200 (ex. FOR J = 1 TO 120),

ligne 200 : on recommence la boucle si I n'est pas à 20,

ligne 210 : retour au début de la boucle pour refaire les 20 séquences lumineuses.

Pour sortir des séquences lumineuses, il faudra appuyer sur la barre d'espacement.

⑥ Commande d'un relais par la carte 8 sorties

Les sorties de la carte huit sorties peuvent débiter assez pour commander directement des Relais type REED (impédance environ 300 Ω sous 5 V) (*fig.* 6.1).

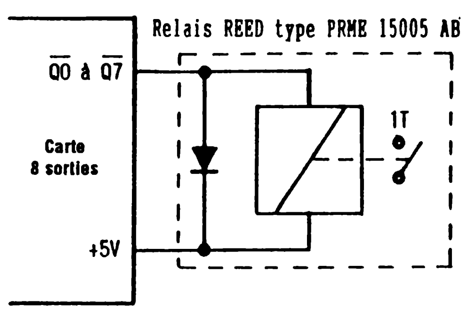


Fig. 6.1

Ces relais comportent juste un contact 1T pouvant supporter un courant de 200 mA bien suffisant pour commander éventuellement un relais plus puissant (*fig.* 6.2).

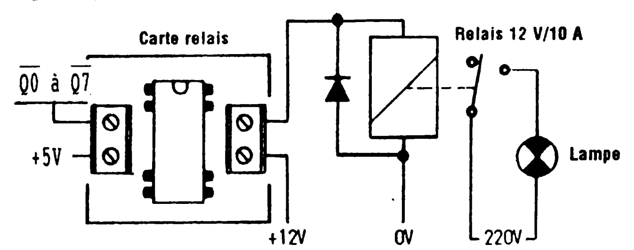


Fig. 6.2

Attention au sens du relais lors du montage. L'encoche doit correspondre avec l'encoche dessinée sur le côté cuivre du circuit. Orienter les borniers, sorties fils vers l'extérieur du circuit (*fig.* 6.3).

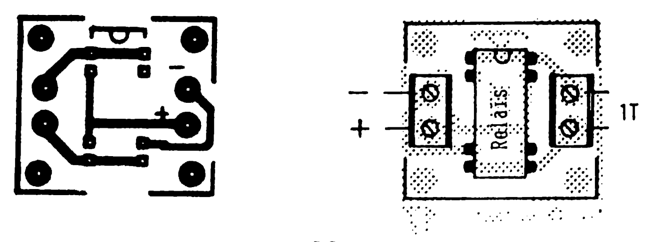


Fig. 6.3

Si vous voulez raccorder la plaquette à la carte huit sorties, il faudra :

— soit se raccorder sur la prise vingt broches en choisissant la sortie désirée,

— soit se raccorder sur un des deux borniers de la plaque huit sorties, mais après avoir au préalable remplacé la résistance 470 Ω correspondant au bornier concerné par une résistance 10 Ω (*fig.* 6.4).

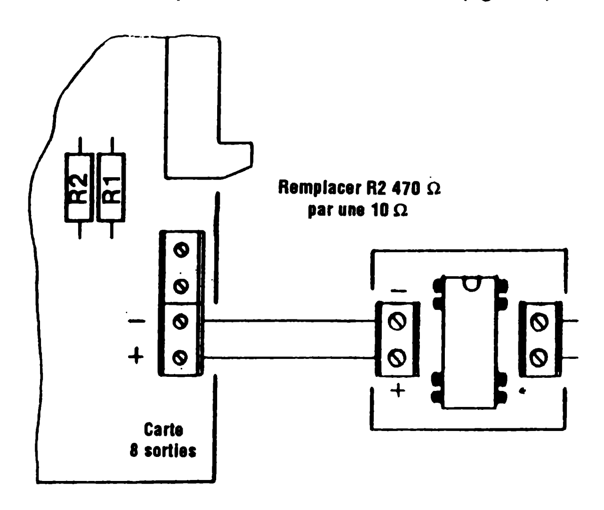


Fig. 6.4

Il suffira de programmer :

OUT &H310,2 pour que le relais colle,
OUT &H310,0 pour que le relais décolle.

Cette deuxième carte relais, munie d'un optocoupleur, se connecte sur la carte huit sorties par l'intermédiaire des deux borniers sans changer la résistance de 470 Ω (*fig.* 6.5).

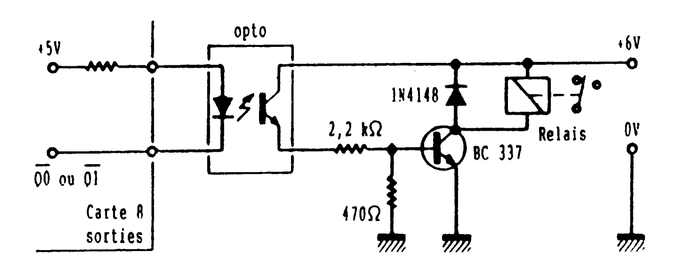


Fig. 6.5

Attention au sens de l'opto repéré par un point et celui de la diode repéré par une bague. Orienter les borniers sorties fils vers l'extérieur du circuit.

Câbler le bornier marqué entrée sur l'un des borniers $\overline{Q0}$ ou $\overline{Q1}$ de la carte huit sorties en respectant la polarité + et –. Amener une alimentation extérieure continue de 6 V sur le bornier + 6 V et 0 V. On dispose sur le troisième bornier d'un contact travail avec un pouvoir de coupure de 10 A.

Pour faire coller le relais, programmer OUT &H310,1 si la carte est branchée entre $\overline{Q0}$ et le plus et OUT &H310,2 si elle est branchée entre $\overline{Q1}$ et le plus. La commande OUT &H310,0 fera décoller le relais dans les deux cas (*fig.* 6.6).

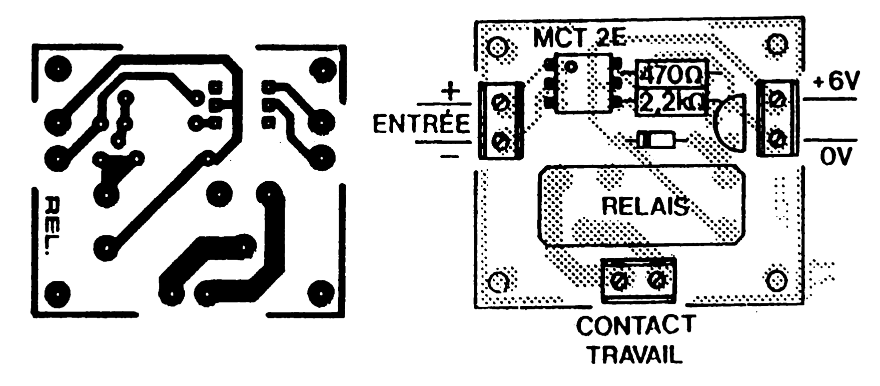


Fig. 6.6

Liste des composants

1 x 470 Ω (jaune violet marron),
1 x 2,2 Ω (rouge rouge rouge),
1 x 1N 4148,
1 x BC 337-25,
1 x opto MCT 2E,
1 x relais 6 V/1RT/10 A,
3 x borniers deux plots.

Les deux cartes relais peuvent être utilisées avec les programmes TRIAC1 (page 00) et **simule** (page 00).

⑦ Carte TRIAC

PRINCIPE

Grâce à l'emploi d'un optotriac, on commande directement la gâchette d'un triac tout en assurant une isolation parfaite de l'ordinateur par rapport au secteur (*fig.* 7.1).

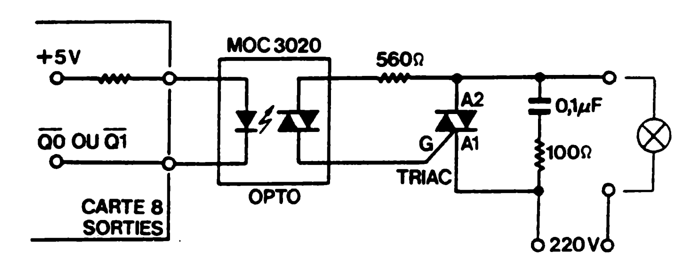


Fig. 7.1

Lorsque la sortie $\overline{Q0}$ *ou* $\overline{Q1}$ passe à 0 V (par OUT &H310,1 pour $\overline{Q0}$ et OUT &H310,2 pour $\overline{Q1}$) la diode LED de l'optotriac émet, débloquant le triac par sa gâchette et allumant la lampe en sortie. Le retour de $\overline{Q0}$ et $\overline{Q1}$ à + 5 V par OUT &H310,0 rebloque le triac et éteint la lampe.

La commande de deux cartes triac se fait simultanément en se servant des deux sorties $\overline{Q0}$ et $\overline{Q1}$ par la commande : OUT &H310,3 ($\overline{Q0}$ =1; $\overline{Q1}$=2; $\overline{Q0}$+$\overline{Q1}$=3).

Le circuit RC (condensateur 0,1 µF et résistance de 100 Ω) évite les montées brusques en tension lors de la commutation du triac.

Il faudra bien faire attention à ce que le 220 V ne vienne pas toucher à la partie ordinateur du circuit (*fig.* 7.2).

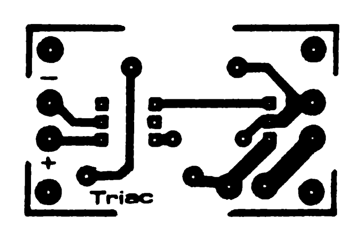


Fig. 7.2

Vérifier l'isolement avant de mettre sous tension.

Attention au sens du triac repéré par le coin tronqué. Le MOC 3020 est repéré par un point ou une encoche. Orienter les borniers pour présenter leurs sorties vers l'extérieur du circuit.

Relier le bornier marqué entrée sur le bornier $\overline{Q0}$ de la carte huit sorties en respectant la polarité + et –.

Brancher une lampe sur le bornier de sortie (maximum 200 W).

Amener le fil secteur sur le bornier repéré 220 V (*fig.* 7.3).

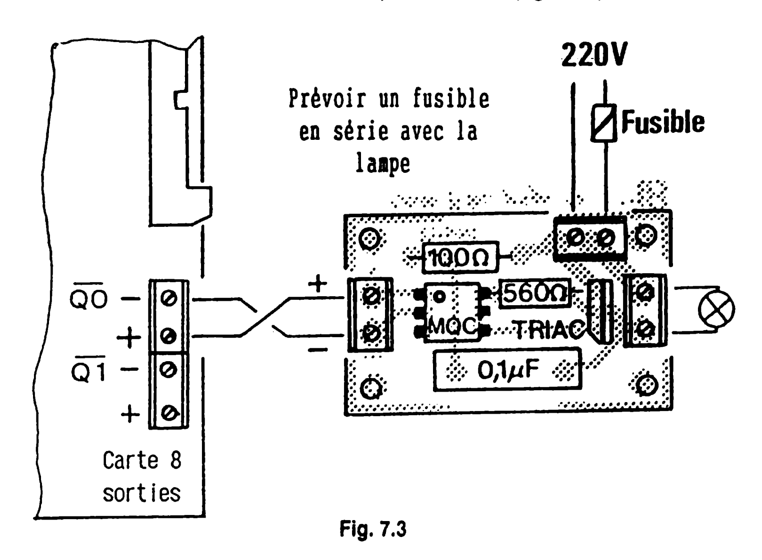


Fig. 7.3

Liste des composants

1 × 100 Ω/1 W (marron noir marron),
1 × 560 Ω (vert bleu marron)
1 × 0,1 µF/250 V,
1 × triac ZO409,
1 × MOC 3020,
3 × bornier de deux plots.

Grâce à la carte triac dont l'entrée est branchée entre $\overline{Q0}$ et + (bornier du haut), le programme TRIAC1 fait clignoter une ampoule.

PROGRAMME

Sous GW BASIC, composer RUN"TRIAC1" puis entrée (*enter*).

```
10 '---triac1------
20 CLS
30 LOCATE 10,20:PRINT" Retour au BASIC appuyer sur barre espace "
40 LOCATE 1,8
50 PRINT" Clignotement par TRIAC. Appuyer sur + plus vite ,- moins vite "
60 TOT=100
70  FOR I=1 TO TOT
80 OUT &H310,1
90 A$=INKEY$
100 IF A$=" " THEN CLS:OUT &H310,0:END
110 IF A$="-" THEN TOT=TOT+3
120 IF TOT<10 THEN GOTO 140
130 IF A$="+" THEN TOT=TOT-3
140 NEXT I
150  FOR I=1 TO TOT
160 OUT &H310,0
170 A$=INKEY$
180 IF A$=" " THEN CLS:END
190 IF A$="-" THEN TOT=TOT+3
200 IF TOT<10 THEN GOTO 220
210 IF A$="+" THEN TOT=TOT-3
220 NEXT I
230 GOTO 70
```

EXPLICATION DU PROGRAMME

ligne 20 : RAZ écran,

lignes 30 à 50 : affichage texte,

ligne 60 : constante nombre de fois boucle TOT=100,

ligne 70 : de 1 à valeur TOT : boucle,

ligne 80 : allumage lampe,

ligne 90 : entrer dans A$,

ligne 100 : test si A$=" " (barre espace), RAZ écran, éteindre la lampe et retour au basic,

ligne 110 : test si A$=– augmentation de TOT de 3 (augmente le nombre de boucles donc ralentit le clignotement),

ligne 120 : test si TOT<10 on ne diminue plus le nombre de boucles, aller à 140,

ligne 130 : test si A$=+ (et TOT>10) TOT=TOT-3 (on diminue le nombre de boucles donc on accélère le clignotement),

ligne 140 : retour à 70 avec incrémentation de I si I n'est pas égal à TOT,

lignes 150 à 220 : idem à 70 à 140 mais avec la lampe éteinte,

ligne 230 : retour à 70.

Le retour au basic se fait en appuyant sur la barre d'espacement. En appuyant sur plus, on accélère le clignotement et sur – on le ralentit.

⑧ Simulation de présence

Ce programme permet, à l'aide de la carte commande 1 Triac, d'allumer et d'éteindre une lampe aux heures que vous aurez choisies (dix séquences allumage extinction au maximum).

Après avoir chargé GW BASIC, faire RUN"SIMULE". Le programme vous affiche l'heure (si l'heure est fausse, remettre à l'heure : voir fin du programme) et vous demande de composer la 1re heure à laquelle vous désirez que la lampe s'allume (composer sous la forme HHMMSS) : par exemple, 22 h 35 min 12 s, composer 223512, puis faire entrée (enter).

Heure d'extinction : composer de même l'heure à laquelle la lampe devra s'éteindre. Par exemple, 22 h 38 min 00 s, composer 223800, puis faire entrée (*enter*).

Le programme vous demande l'heure d'allumage pour la séquence 2. Si vous composez 0 puis entrée (enter), le programme passe en phase de commande. L'heure clignote en haut à gauche et les secondes défilent.

L'heure de début de la phase commande reste allumée au milieu de l'écran et vous signale que la lampe est éteinte.

Appuyer sur barre espace pour revenir au BASIC.

Refaire RUN.

Composer la première heure d'allumage en ajoutant 1 minute à l'heure affichée en haut à gauche.

Exemple : heure affichée : 10:37:22, composer 103822, puis entrée (enter).

Heure d'extinction affichée + 2 minutes, soit dans notre exemple 103922, puis entrée (*enter*).

Composer heure séquence 2, 0 puis entrée (*enter*).

L'heure s'affiche en haut à gauche. Lorsqu'elle atteindra la valeur de l'heure d'allumage, la lampe sera commandée et on affichera au milieu de l'écran l'heure d'allumage suivit de « allumée », 1 minute plus tard, la lampe s'éteindra, et on affichera « éteinte ». Vous pouvez entrer jusqu'à 10 séquences allumée éteinte. Lorsque l'on compose l'heure d'allumage à zéro, on passe en phase de commande de la lampe. De même, après la composition de la 10e séquence, on passe automatiquement en phase de commande de la lampe. Vérifier bien que les heures entrées soient possibles. Ne pas dépasser 23.59.59. Par exemple un affichage 45 20 18 ne donnera rien de même qu'un affichage négatif.

Appuyer sur barre espace pour revenir au basic.

REMISE À L'HEURE DE L'ORDINATEUR

Composer sous GW BASIC

PRINT TIME$ puis entrée (enter), l'ordinateur vous affiche l'heure. Si celle-ci est erronée, composer TIME$ = "HH:MM:SS" puis touche entrée (enter).

Recomposer PRINT TIME$ pour vérifier que l'heure a bien été modifiée.

Si vous arrêtez le programme par CONTROL.C ou CONTROL.PAUSE votre écran peut garder un fond blanc et le curseur être invisible (commande exécutée dans le programme).

Appuyer sur Echap (ESC), puis composer COLOR 7,0 puis entrée (*enter*). Un OK sur fond noir doit apparaître. Composer CLS puis entrée (*enter*) pour remettre l'écran en noir.

Pour récupérer le curseur, appuyer sur Echap (ESC) une ligne noire doit apparaître, puis composer LOCATE 1,1,1, puis touche entrée (*enter*). En haut de l'écran, un OK doit apparaître en dessous duquel le curseur clignote.

PROGRAMME SIMULÉ

```
10 '-----programme---simule-présence--
20 KEY OFF
30 CLS
40 PRINT TIME$
50 LOCATE 10
60 FOR I=1 TO 10
70 LOCATE ,10:PRINT USING"##";I;
80 PRINT"    Heure allumer    ";:INPUT ;A(I)
90 IF A(I)=0 THEN LOCATE ,10:PRINT SPC(40):GOTO 130
100 LOCATE,42
110 PRINT"    Heure éteindre    ";:INPUT B(I)
120 NEXT I
130 LOCATE 5,31,0:COLOR 0,7
140 OUT &H310,0:PRINT " "; TIME$;" : Eteinte ";
150 LOCATE 25,20:PRINT" Retour au BASIC appuyer sur barre espace . ";
160 A=VAL(LEFT$(TIME$,2)+MID$(TIME$,4,2)+RIGHT$(TIME$,2))
170 B$=TIME$
180 LOCATE 1,1
190 COLOR 31,0
200 PRINT B$;
210 COLOR 0,7
220 FOR I=1 TO 10
230 LOCATE 5,31
240 IF A=A(I) THEN OUT &H310,1:PRINT " "; B$;" : Allumée ";
250 LOCATE 5,31
260 IF A=B(I) THEN OUT &H310,0:PRINT " "; B$;" : Eteinte ";
270 NEXT I
280 A$=INKEY$
290 IF A$=" " THEN COLOR 7,0:CLS:OUT &H310,0:LOCATE 1,1,1:END
300 GOTO 160
```

EXPLICATION DU PROGRAMME

ligne 20 : éteindre ligne 25,

ligne 30 : RAZ écran,

ligne 40 : afficher l'heure,

ligne 50 : à partir de la ligne 10,

ligne 60 : pour 10 séquences numérotées dans I,

ligne 70 : en position 10, afficher valeur de I,

ligne 80 : afficher heure allumer et entrer dans A(I) (soit A(1) à ... A(10),

ligne 90 : si A(I) = 0 (heure entrée = 0), effacer la ligne que l'on vient de visualiser et aller à 130,

ligne 100 : en position 42 de la ligne,

ligne 110 : afficher heure éteindre et l'entrer dans B(1) à ... B(10),

ligne 120 : retourner à 60 si boucle n'est pas égale à 10,

ligne 130 : boucle = à 10 ou composition heure allumée = 0 en ligne 90. A la ligne 5 en position 31 en éteignant le curseur, couleur inversé,

ligne 140 : éteindre la lampe. Afficher l'heure suivi de éteinte,

ligne 150 : en ligne 25 position 20 afficher « retour au basic appuyer sur la barre d'espacement »,

ligne 160 : mettre l'heure sur forme HHMMSS dans A,

ligne 170 : mettre l'heure dans B$,

ligne 180 : ligne 1 position 1,

ligne 190 : affichage clignotant,

ligne 200 : afficher l'heure,

ligne 210 : fond d'affichage inversé,

ligne 220 : pour 10 séquences numérotées dans I,

ligne 230 : ligne 5 position 31,

ligne 240 : si heure = heure dans A(I) afficher heure suivi d'allumée,

ligne 250 : ligne 5 position 31,

ligne 260 : si heure = heure dans B(I) afficher heure suivi d'éteinte,

ligne 270 : retour à 220 si boucle n'est pas égale à 10,

ligne 280 : entrée dans A$,

ligne 290 : si A$ = " " (barre espace enfoncée) écran fond noir, extinction de la lampe, rallumer le curseur, retour au basic,

ligne 300 : retour à 160.

⑨ Carte PIA

TECHNOLOGIE

L'interface PIA/PC est constituée autour d'un circuit intégré 8255, associé à un circuit de décodage d'adresse, ainsi qu'un circuit buffer.

Trois ports de huit données chacun (1 octet) sont disponibles.

L'interface pourra être configurée soit en 16 entrées 8 sorties, soit en 8 entrées 16 sorties. Huit des sorties sont fixées (port A) et amplifiées.

COMPATIBILITÉ ET CONNECTION

PIA/PC peut être connectée à tous les ordinateurs compatible PC : PC/XT, PC/AT.

L'interface est à enficher dans l'un des connecteurs libres (8 Bits) de votre ordinateur.

Il y a possibilité également de mettre deux cartes PIA dans le même ordinateur grâce à un jeu de straps sélectionnant deux jeux de différentes adresses.

PÉRIPHÉRIQUES

L'interface PIA/PC 24 entrées/sorties peut être reliée grâce à un câble nappe aux périphériques de la gamme :

— chenillard,

— carte 4 entrées/4 sorties,

— programmateur d'EPROM,

— circuit spécial de test de la carte PIA,

— ou périphérique de votre choix en respectant le brochage.

SCHÉMA DE PRINCIPE

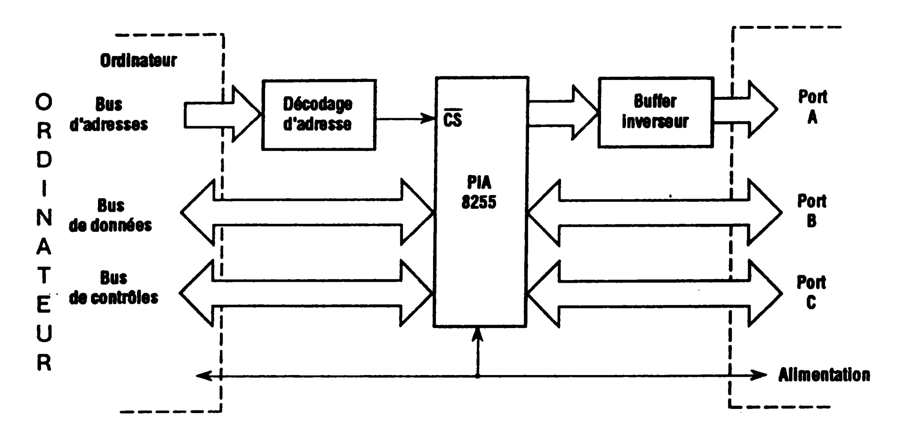


Fig. 9.1

DÉCODAGE DE L'ADRESSE

Le 8255 ne peut recevoir ou envoyer des données au micro que lorsque son entrée CS (chips select) est activée (niveau bas : 0). Le décodage d'adresse (74 LS 04 et 74 LS 133) permet d'obtenir ce niveau seulement pour des adresses bien précises.

La zone mémoire réservée pour les cartes connectables de votre PC se situe de l'adresse (hexadécimal) : 300(H) à 31F(H).

Seules quatre adresses sont nécessaires pour commander le PIA 8255.

Un strap vous permettra de sélectionner quatre adresses basses (strap en A), ou quatre adresses hautes (strap en B).

Vous pourrez donc ainsi commander deux cartes PIA (l'une avec strap en A l'autre avec strap en B).

	Strap en A adresses basses	Strap en B adresses hautes
Adresse port A	&H310	&H314
Adresse port B	&H311	&H315
Adresse port C	&H312	&H316
Adresse registre de contrôle	&H313	&H317

	Bits du bus d'adresse	**A9**	**A8**	**A7**	**A6**	**A5**	**A4**	**A3**	**A2**	**A1**	**A0**
Strap en A	*code binaire*	1	1	0	0	0	1	0	0	X	X
	code hexadécimal	3		1				0 à 3			
Strap en B	*code binaire*	1	1	0	0	0	1	0	1	X	X
	code hexadécimal	3		1				4 à 7			

Adresse d'E/S	Fonction
000_H...$00F_H$	Contrôleur DMA (8237A-5)
020_H...021_H	Contrôleur d'interruption (8259-5)
040_H...043_H	Temporisateur/compteur (8253-5)
060_H...063_H	Registre du système (8255A-5)
080_H...083_H	Registre de pages DMA (74LS670)
$0A0_H$...$0BF_H$	Registre d'interruption NMI
$0C0_H$...$0FF_H$	Réservé
100_H...$1FF_H$	Contrôleur de disque dur
200_H...$20F_H$	Port de manche de commande (jeux)
210_H...217_H	Cartes d'extension
220_H...$24F_H$	Réservé
278_H...$27F_H$	Seconde imprimante
$2F8_H$...$2FF_H$	Seconde interface sérielle
300_H...$31F_H$	Cartes prototypes
320_H...$32F_H$	Contrôleur de disque dur
378_H...$37F_H$	Interface imprimante (parallèle)
380_H...$38F_H$	Interface SDLC
$3A0_H$...$3AF_H$	Réservé
$3B0_H$...$3BF_H$	Adaptateur monochrome et imprimante
$3C0_H$...$3CF_H$	Réservé
$3D0_H$...$3DF_H$	Carte graphique
$3E0_H$...$3E7_H$	Réservé
$3F0_H$...$3F7_H$	Interface de lecteur de disquettes
$3F8_H$...$3FF_H$	Interface sérielle

PROGRAMMATION DU PIA 8255

Pour pouvoir configurer le PIA 8255, il faut accéder au registre de contrôle (définit à l'adresse &H313 ou &H317), puis charger les bits de ce registre avec une valeur bien définie.

CHARGEMENT DU REGISTRE DE CONTRÔLE

Les bits du registre de contrôle définis à l'adresse &H313 (ou &H317) vous permettent d'actionner le PIA (bit 7), de sélectionner le mode (bits 6,5) et de commander les trois ports en entrée ou en sortie (bits 0, 1, 2, 3, 4).

Le mode 0 sera principalement utilisé pour plus de simplicité. Pour d'autres modes, se reporter à la documentation du 8255.

Bit du registre de contrôle (*fig.* 9.2)

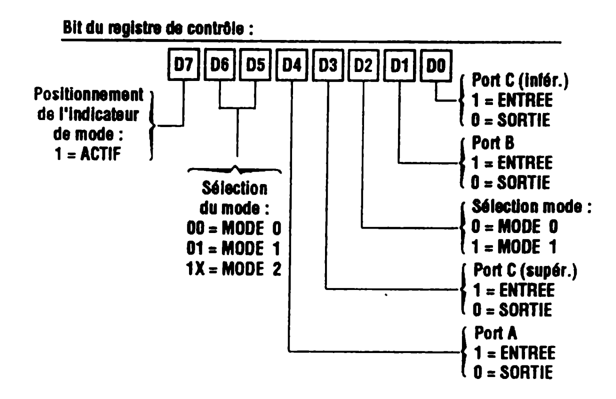


Fig. 9.2

MODE 0 : Les ports A et B et les deux parties (inférieure et supérieure) du port C peuvent être programmés en ENTRÉES ou en SORTIES.

MODE 1 : Chacun des ports A et B est servi par un demi port C, chargé de fournir les signaux d'aquittement.

MODE 2 : Le port A est intégralement bidirectionnel, mais pour cela, il doit être servi par 5 des lignes du port C.

Exemples

Configuration du PIA	Bit du registre de contrôle								
	D7	D6	D5	D4	D3	D2	D1	D0	*Valeur hexadécimale*
Activation du mode Port A en sortie Port B et C en entrée Mode 0	1	0	0	0	1	0	1	1	&H8B
Activation du mode Port A et B en sortie Port C en entrée Mode 0	1	0	0	0	1	0	0	1	&H89
Activation du mode Port A et C en sortie Port B en entrée Mode 0	1	0	0	0	0	0	1	0	&H82

EXEMPLE DE PROGRAMMATION DU PIA 8255 EN BASIC

(avec strap en A)

```
10 OUT &H313, &H8B
20 OUT &H310, 32
30 B = INP (&H311)
40 C = INP (&H312)
```

EXPLICATION PROGRAMME

ligne 10 : on charge à l'adresse du registre de contrôle la valeur hexadécimale 8B, donc le port A est en sortie et les ports B et C sont en entrées.

ligne 20 : on charge à l'adresse du Port A la valeur binaire codée décimale (BCD) : 32 donc $\overline{PA5}$ passe à 0 niveau bas (il y a inversion binaire, car le Port A dispose d'un circuit buffer inverseur).

Exemple : en chargeant le Port A avec la valeur BCD:255, tous les bits du port A seront au niveau bas (0).

ligne 30 : la valeur (BCD) présente sur le Port B est chargée grâce à l'instruction INP dans « B ».

Exemple : si le bit PB6 est à « 1 », les autres bits du port B restant au niveau bas, la valeur lue dans B sera : 64.

ligne 40 : identique à la ligne 30, sauf qu'il s'agit du port C qui est chargé dans « C ».

Exemple ligne 20

Valeur BCD	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Bits du Port A	$\overline{\text{PA7}}$	$\overline{\text{PA6}}$	$\overline{\text{PA5}}$	$\overline{\text{PA4}}$	$\overline{\text{PA3}}$	$\overline{\text{PA2}}$	$\overline{\text{PA1}}$	$\overline{\text{PA0}}$
Valeur binaire du port de sortie	1	1	0	1	1	1	1	1

PORTS ENTRÉES/SORTIES

Ports d'entrées : seuls les ports B et C peuvent être utilisés en entrée. Un niveau logique haut (1) est établi par une tension positive de + 5 V sur l'une des entrées. Un niveau logique bas (0) est réalisé par une tension nulle : 0 V.

Ports de sorties :

— *Port A :* ce port sera toujours configuré en sortie, car les bits PA0 à PA7 disposent d'un circuit spécial buffer inverseur (à collecteur ouvert) ULN 2803 (*fig.* 9.3).

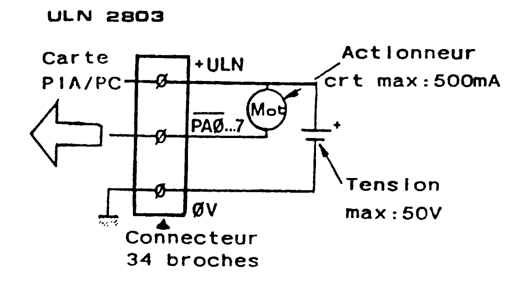


Fig. 9.3

Ce circuit contient un réseau de huit transistors darlingtons supportant un $V_{ce\,max}$ de 50 V et un courant $I_{c\,max}$ de 500 mA.

Ce circuit vous permet d'alimenter directement un actionneur (moteur, pompe...) grâce à une alimentation continue externe.

— *Ports B et C :* les ports B et C peuvent être programmés également en sortie. Dans ce cas, le bit de sortie activé au niveau logique haut ne peut fournir qu'un courant de 1 mA sous une tension d'environ 5 V. Un niveau logique bas fourni une tension nulle (*fig.* 9.4).

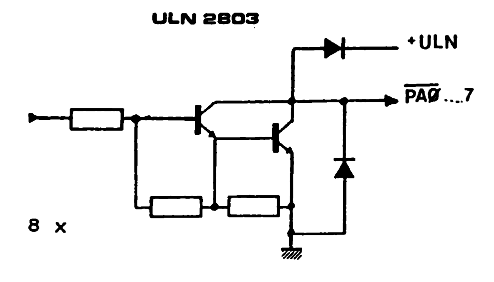


Fig. 9.4

CARTE PIA (*fig.* 9.5)

BROCHAGE (*fig.* 9.6 et 9.7)

Signal	Numérotation			Signal
	Côté pistes		Côté composants	
GND	B01		A01	
Reset	B02		A02	D7
	B03		A03	D6
	B04		A04	D5
	B05		A05	D4
	B06		A06	D3
	B07	côté du boîtier PC	A07	D2
	B08		A08	D1
	B09		A09	D0
	B10		A10	
	B11		A11	AEN
	B12		A12	
IOWC	B13		A13	
IORC	B14		A14	
	B15		A15	
	B16		A16	
	B17		A17	
	B18		A18	
	B19		A19	
	B20		A20	
	B21		A21	
	B22		A22	A9
	B23		A23	A8
	B24		A24	A7
	B25		A25	A6
	B26		A26	A5
	B27		A27	A4
	B28		A28	A3
+ 5 V	B29		A29	A2
	B30		A30	A1
GND	B31		A31	A0

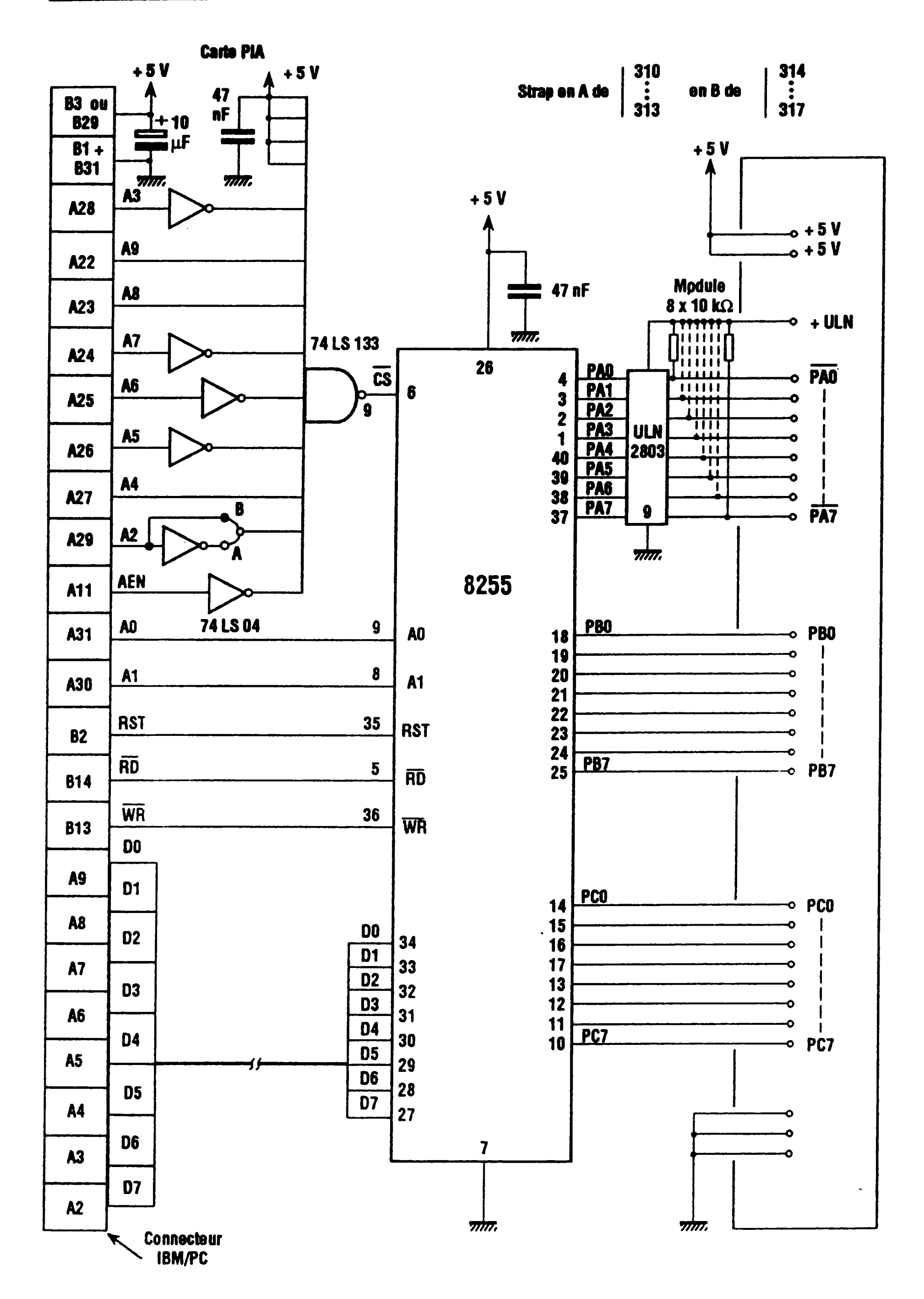
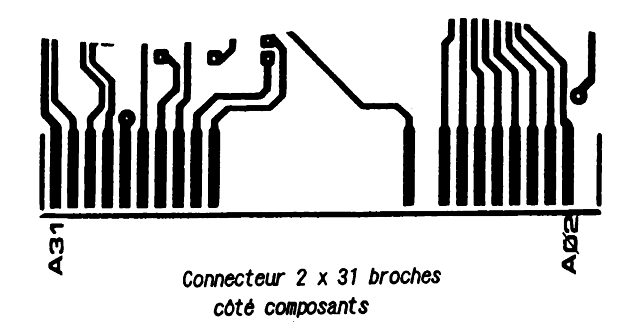
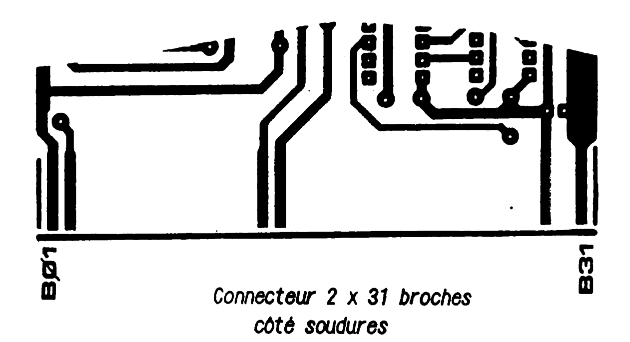

Carte PIA
+5 V
+5 V
47 nF
+10 µF
B3 ou B29
B1 + B31
A28
A22
A23
A24
A25
A26
A27
A29
A11
A31
A30
B2
B14
B13
A9
A8
A7
A6
A5
A4
A3
A2
A3
A9
A8
A7
A6
A5
A4
A2
B
A
AEN
74 LS 04
74 LS 133
CS
6
9
A0
A1
RST
RD
WR
D0
D1
D2
D3
D4
D5
D6
D7
8255
26
7
Strap en A de 310 313
en B de 314 317
+5 V
47 nF
Module 8 x 10 kΩ
+ ULN
ULN 2803
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB0
PB7
PC0
PC7
Connecteur IBM/PC

Fig. 9.5



Connecteur 2 x 31 broches
côté composants



Connecteur 2 x 31 broches
côté soudures

Fig. 9.6

MONTAGE DU CIRCUIT DOUBLE FACE

Montage du circuit :

— commencer par souder les deux traversées du circuit. Pour cela, utiliser des queues de composants; elles sont repérées par la lettre X sur la figure 1,

— souder les quatre straps, ainsi que le strap en A ou B selon les adresses choisies. Ne pas oublier la soudure côté composants de trois straps (*fig.* 9.8),

— souder les condensateurs, ne pas oublier la soudure côté composants de certains condensateurs,

— souder le réseau de résistances (réf. 709 A 103) en faisant attention de bien le positionner,

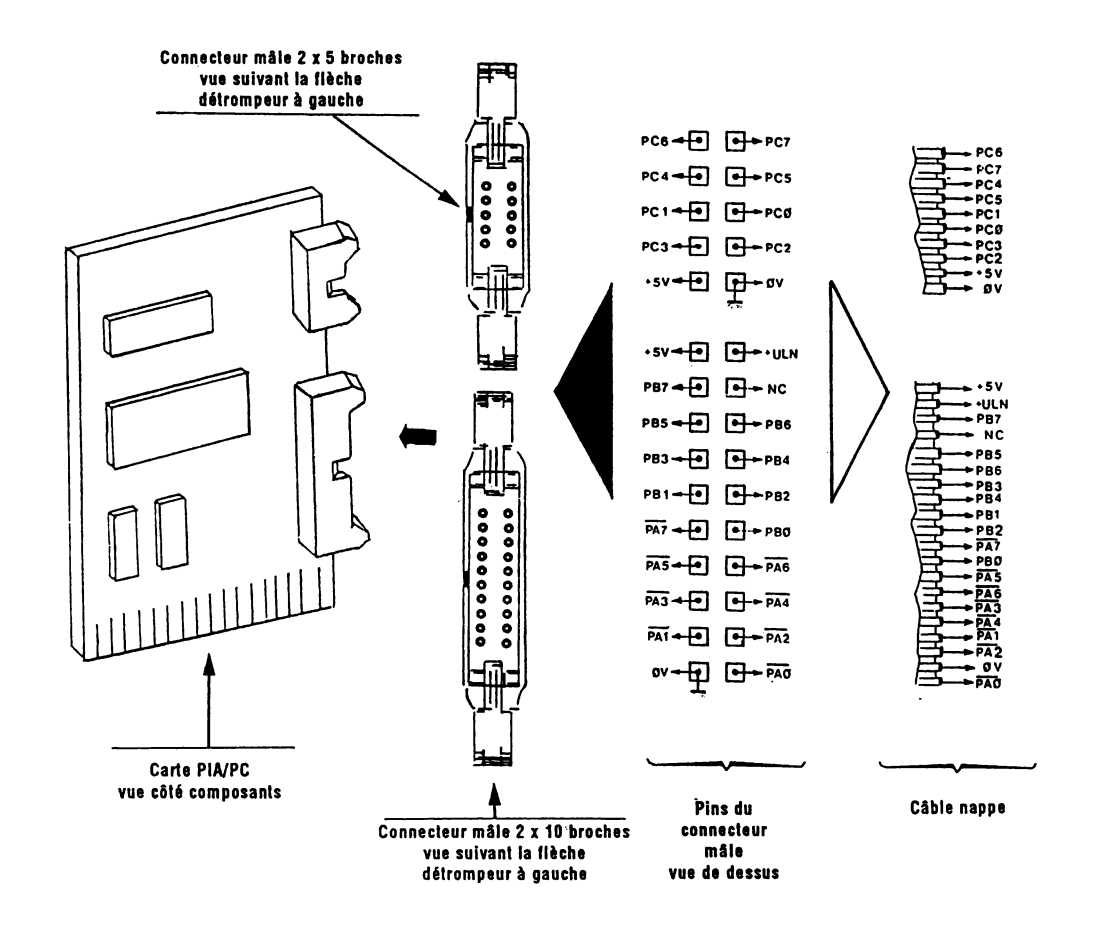

Connecteur mâle 2 x 5 broches vue suivant la flèche détrompeur à gauche
Câble nappe
Pins du connecteur mâle vue de dessus
Connecteur mâle 2 x 10 broches vue suivant la flèche détrompeur à gauche
Carte PIA/PC vue côté composants

Fig. 9.7

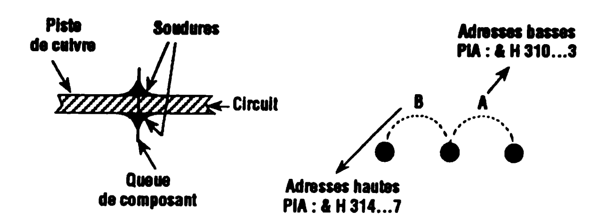


Fig. 9.8

— souder les circuits intégrés en positionnant leurs encoches d'après la figure 9.10. Ne pas oublier les soudures des pattes côté composants,

— souder les connecteurs mâles coudés côté composants.

CIRCUIT IMPRIMÉ CÔTÉ CUIVRE

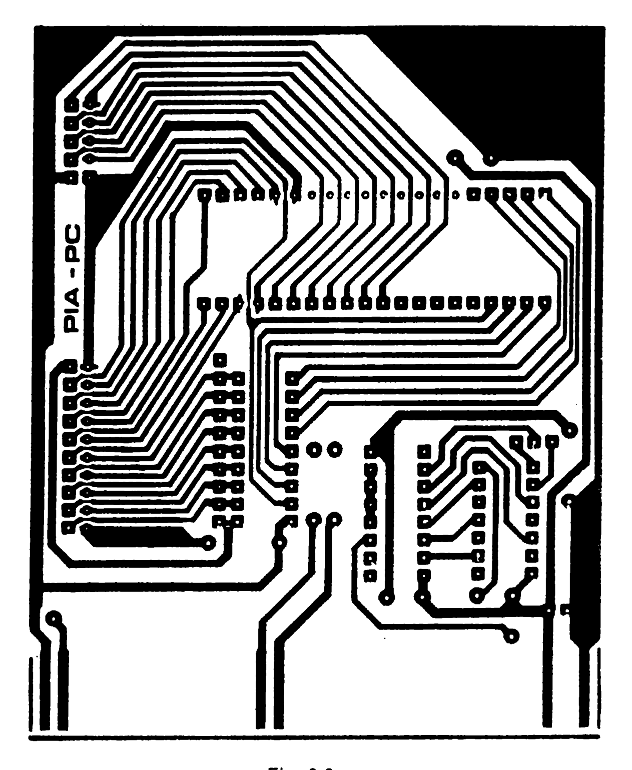


Fig. 9.9

IMPLANTATION DES COMPOSANTS

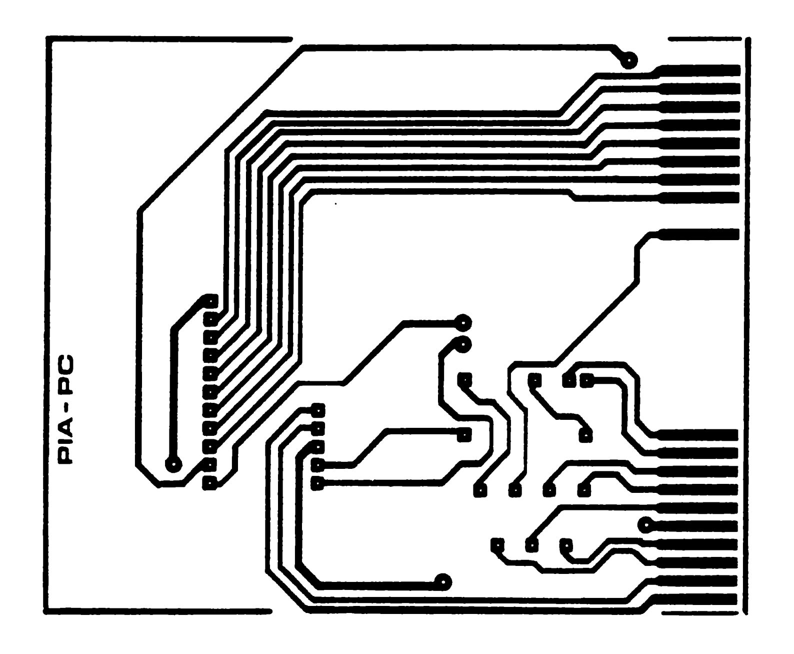


Fig. 9.9 bis – carte PIA-IC - côté composants

Liste des composants

PIA 8255,
74 LS 04,
74 LS 133,
ULN 2803 A,
3 × 22 à 100 nF,
10 µF/20 V tantale goutte,
709 ou 710 A 103 réseau,
Circuit double face percé,
Connecteur mâle coudé 2 × 10 broches,
Connecteur mâle coudé 2 × 5 broches,
Un câble informatique 20 conducteurs,
Un câble informatique 10 conducteurs.

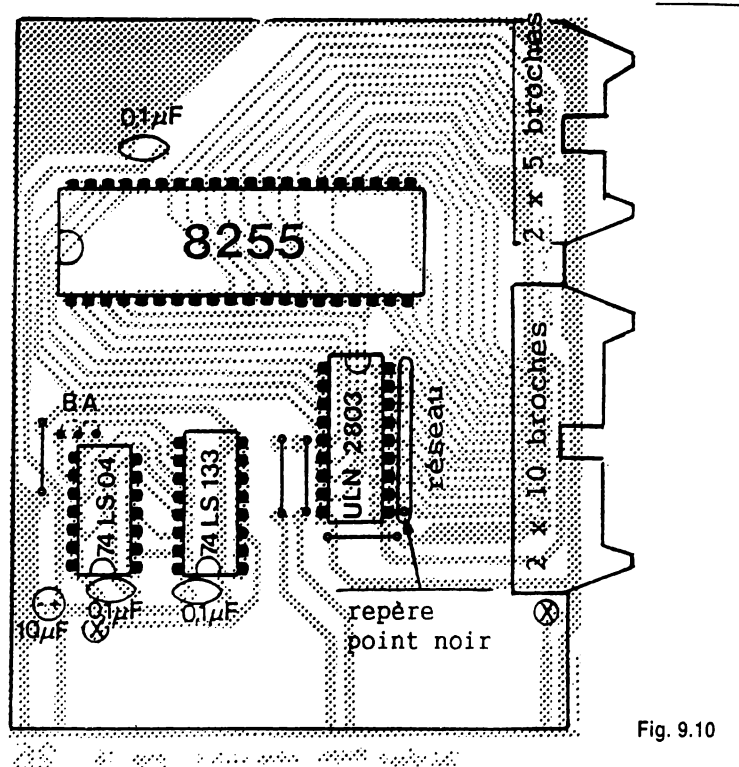


Fig. 9.10

MONTAGE DE LA CARTE PIA/PC DANS VOTRE COMPATIBLE

Après avoir débranché celui-ci, il faudra retirer le capot ou carter supérieur (éventuellement voir notice constructeur).

Insérer la carte PIA/PC dans un des connecteurs d'extensions de 2 × 31 broches en vérifiant le sens d'insertion à l'aide du manuel constructeur. Retirer un des caches métalliques en vis-à-vis avec le connecteur que l'on utilise pour faire passer le câble nappe.

Remarques importantes

— Respecter impérativement le sens d'insertion de la carte PIA/PC dans l'ordinateur (risque de destruction).

— Pour mettre en route le système informatique, alimenter au préalable les périphériques (écran, imprimante, carte 4 entrées/4 sorties...), puis mettre l'ordinateur sous tension.

— Pour mettre hors tension, couper en premier l'ordinateur, puis ensuite les périphériques.

⑩ Mini testeur PIA/PC

Cette petite carte permet de tester toutes les combinaisons programmables sur votre PIA/PC (*fig.* 10.1 et *fig.* 10.2).

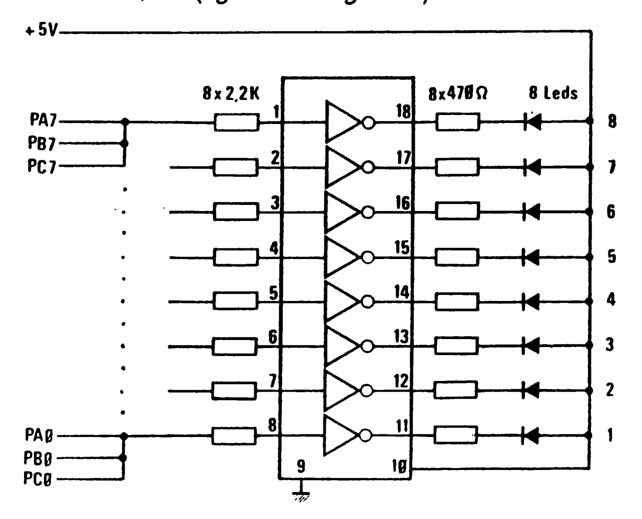


Fig. 10.1

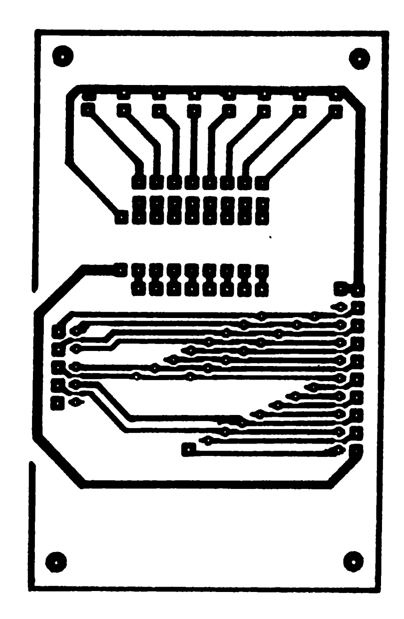


Fig. 10.2

Les trois ports A, B et C sont reliés ensemble :

A1 → B1 → C1, A2 → B2 → C2, etc.

Réunir le mini testeur à la carte PIA avec deux câbles **essais** avec programme ROBOTPC sous DOS (*fig.* 10.3).

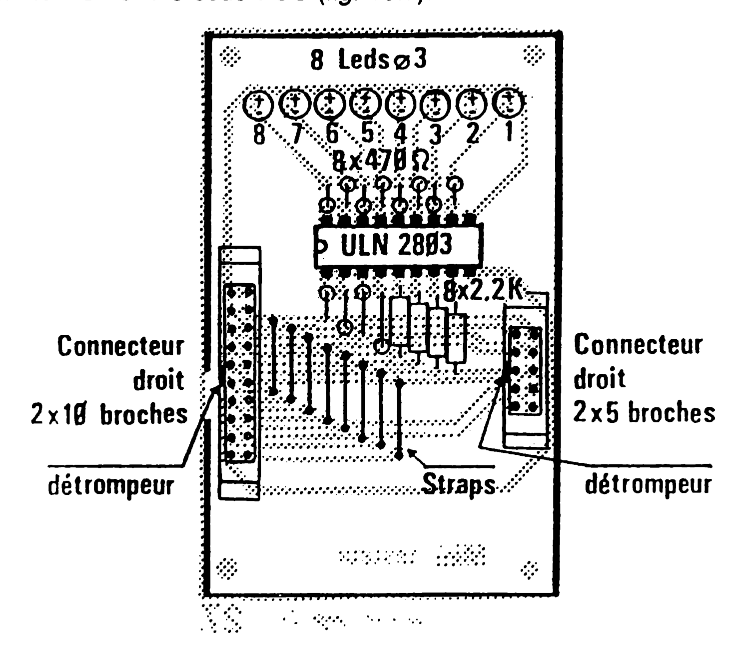


Fig. 10.3

A l'aide de la disquette livrée avec le dossier ou la carte, charger ROBOTPC sous DOS.

A l'apparition du Menu, sélectionner le programme 2, test entrée sortie en composant 2 puis entrée (enter).

Toutes les LED de la carte test doivent s'allumer. En composant 255 puis entrée (enter), elles doivent toutes s'éteindre. Composer 1, seul la LED repérée 1 s'éteind. En effet, le programme à configuré A en sortie et B et C en entrée (*fig.* 10.4).

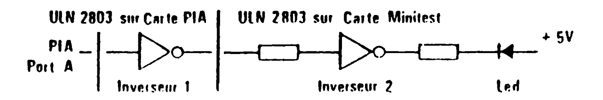


Fig. 10.4

La sortie du port A est reliée à un ULN 2803, c'est-à-dire qu'un plus à l'entrée de l'ULN met sa sortie à 0. Cette sortie étant reliée à l'ULN de la mini test, un 0 à son entrée donne un plus en sortie. La LED étant entre deux plus sera éteinte.

Composer maintenant 254. Seule la LED 1 est allumée. En effet, 254 met toutes les sorties du port A à + sauf la première. 0 sur entrée ULN 1 = + en sortie, + sur entrée ULN 2 = 0 en sortie. La Led entre 0 et + s'allume.

Retour au menu, appuyer sur F2.

Dans le Menu, retour au DOS, composer 4.

PROGRAMME EN BASIC, TEST AVEC MINI CARTE

Programme BASPIA.

Charger GW Basic ou BASICA de Microsoft livré généralement avec le MS DOS de Microsoft.

A l'aide de la disquette du dossier ou de la carte, faire RUN"BASPIA".

```
10 '--programme---BASPIA---
20 COLOR 7,0:CLS
30 KEY OFF
40 LOCATE ,12:PRINT " Test carte pia port a en sortie, b et c en entrée "
50 'config pia-------------
60 P=&H313
70 O=&H310
80 R=&H311
90 T=&H312
100 OUT P,&H8B
110 LOCATE 12,10:PRINT"valeur sortie port a:"
120 LOCATE 12,32 :INPUT A
130 LOCATE 12,32:PRINT"          valeur sortie a=";A;"  "
140 OUT O,A
150 '----test--de--la--valeur--d'entrée--
160 B=INP(R)
170 C=INP(T)
180 LOCATE 15,20:PRINT SPC(15)
190 LOCATE 15,20:PRINT "port b=";B;" port c=";C;"   "
200 GOTO 120
```

EXPLICATION DU PROGRAMME

ligne 20 : RAZ écran,

ligne 30 : effacer ligne 25 de l'écran,

ligne 60 : Port P (configuration) en H313,

ligne 70 : Port A en H310,

ligne 80 : Port B en H311,

ligne 90 : Port C en H312,

ligne 100 : configure A en sortie, B en entrée et C en entrée,

ligne 120 : entrée dans A de la valeur à sortir sur port A,

ligne 140 : valeur sur port A en sortie,

ligne 160 : entrée de la valeur sur port B dans B,

ligne 170 : entrée de la valeur sur port C dans C,

ligne 190 : affichage valeur B et C,

ligne 200 : retour à entrée valeur pour port A.

Composer la valeur que vous voulez en sortie sur port A.

Exemple : 12 puis entrée (*enter*).

Le port B et le port C (reliés ensemble par le mini testeur) reçoivent la valeur 243 (soit le complément à 255 de 12, ce qui est logique, puisque l'ULN de la carte PIA inverse la sortie du port A). Les LED 8 et 4 s'éteignent. En composant 0, B et C recevront 255. En composant 255, B et C recevront 0.

Pour sortir du programme basic, appuyer en même temps sur CTRL et C.

⑪ Carte 4 entrées/4 sorties

Le périphérique 4 Entrées/4 Sorties à pour but de commander des circuits de puissance, ou de recevoir des actions venant de l'extérieur en assurant une parfaite isolation électrique entre l'ordinateur et le milieu environnant.

TECHNOLOGIE

La carte 4 Entrées/4 Sorties est constituée principalement de circuits d'opto-coupleurs (MCT 2E), et de relais de puissances (10 A). Un circuit intégré 4049 ainsi que des transistors NPN assurent une mise en forme correcte des signaux.

Huit voyants permettent de visualiser l'état des entrées ou des sorties.

CONNEXION

— Le périphérique « carte 4 Entrées/4 Sorties » est connectable à l'interface PIA/PC, vingt-quatre entrées sorties (disponible en kit).

— La carte Entrées/Sorties est alimentée par le + 5 V de l'ordinateur, et par une autre alimentation autonome (directement au 220 V) ou externe 12 V continue.

SCHÉMA DE PRINCIPE (*fig.* 11.2)

OPTO-COUPLEUR

Les circuits opto-coupleurs MCT 2E permettent une isolation en tension jusqu'à 2 500 V.

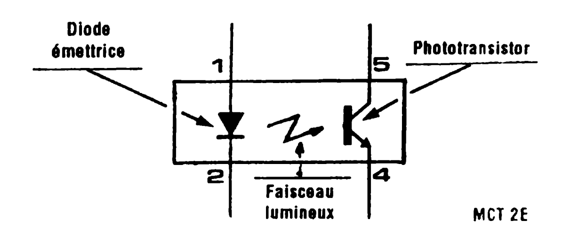


Fig. 11.1

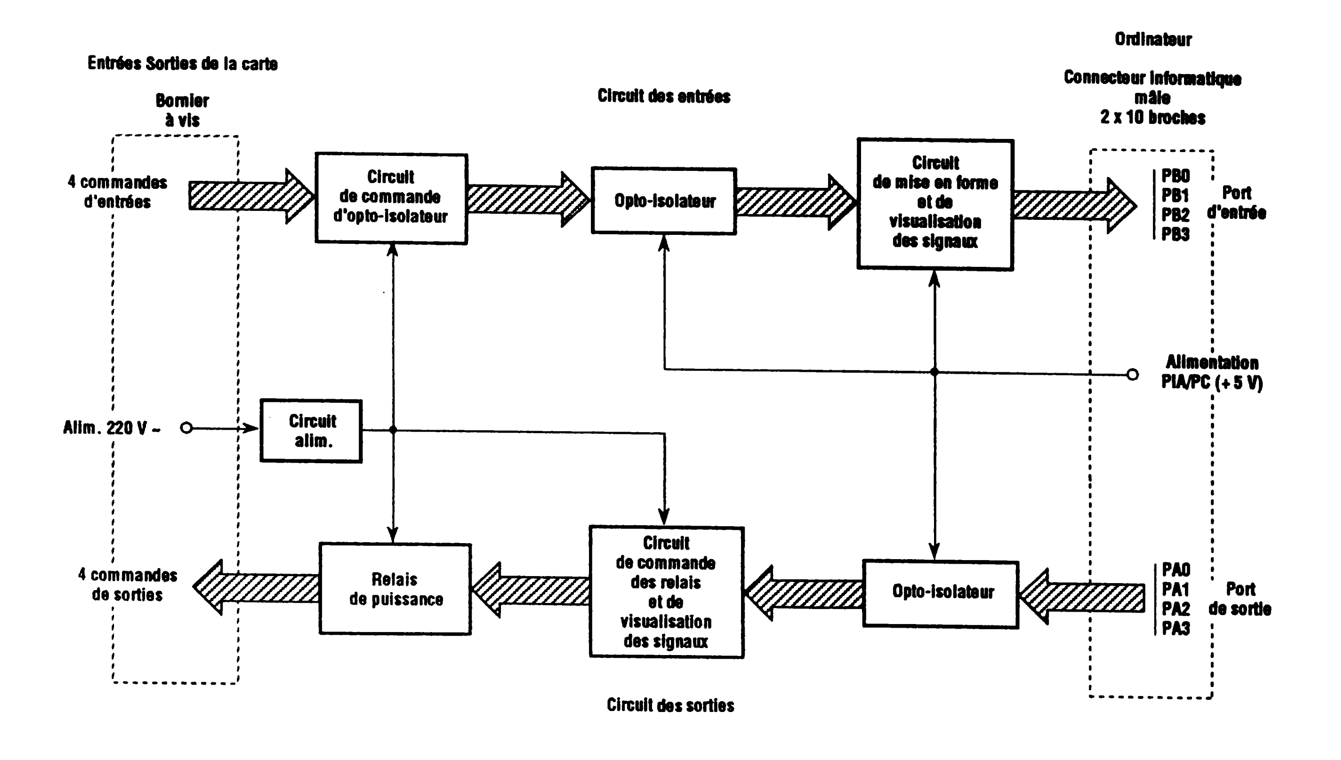

Entrées Sorties de la carte
Bornier à vis
4 commandes d'entrées
Circuit de commande d'opto-isolateur
Circuit des entrées
Opto-isolateur
Circuit de mise en forme et de visualisation des signaux
Ordinateur
Connecteur informatique mâle 2 x 10 broches
PB0
PB1
PB2
PB3
Port d'entrée
Alimentation PIA/PC (+ 5 V)
Alim. 220 V ~
Circuit alim.
4 commandes de sorties
Relais de puissance
Circuit de commande des relais et de visualisation des signaux
Opto-isolateur
PA0
PA1
PA2
PA3
Port de sortie
Circuit des sorties

Fig. 11.2

En faisant circuler un courant dans la diode émettrice (pins 1 et 2), on sature le phototransistor NPN (pins 4 et 6). Sans circulation de courant dans la diode émettrice, le phototransistor est à l'état bloqué.

CIRCUIT D'ENTRÉE

La commande du circuit d'entrée est réalisée par une boucle de courant réglable en intensité (de 227 µA à 3,56 mA), grâce à l'ajustable de 47 kΩ.

Ce signal issu de la boucle de courant est relié à l'entrée d'un circuit buffer inverseur (4049), permettant d'alimenter la diode émettrice de l'opto-coupleur.

La sortie de cet optocoupleur actionne directement l'un des bits (PB0 à PB3) du port d'entrée de la carte PIA/PC, et alimente la LED de visualisation à travers un transistor d'amplification.

CIRCUIT DE SORTIE

Le circuit buffer inverseur (ULN 2803) de la carte PIA/PC permet de commander à travers une résistance de 560 Ω, la diode émettrice de l'opto-coupleur. La sortie de ce circuit actionne à travers un transistor d'amplification la bobine du relais et la LED de visualisation.

CIRCUIT ALIMENTATION

L'alimentation est reliée directement au secteur à travers un transformateur d'isolement. Elle assure un redressement, un filtrage et une régulation de la tension d'alimentation.

SCHÉMA (*fig.* 11.3)

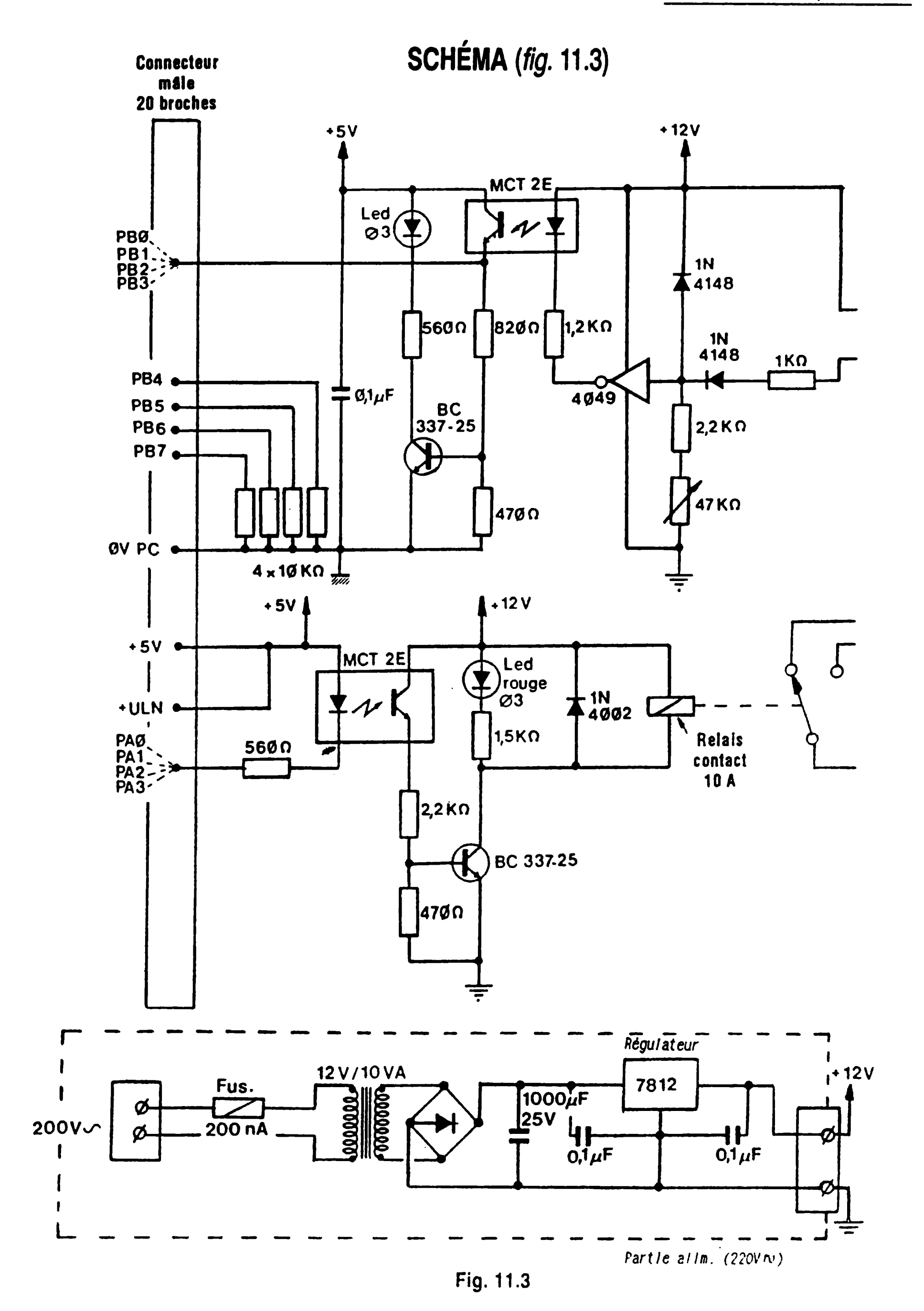


Fig. 11.3

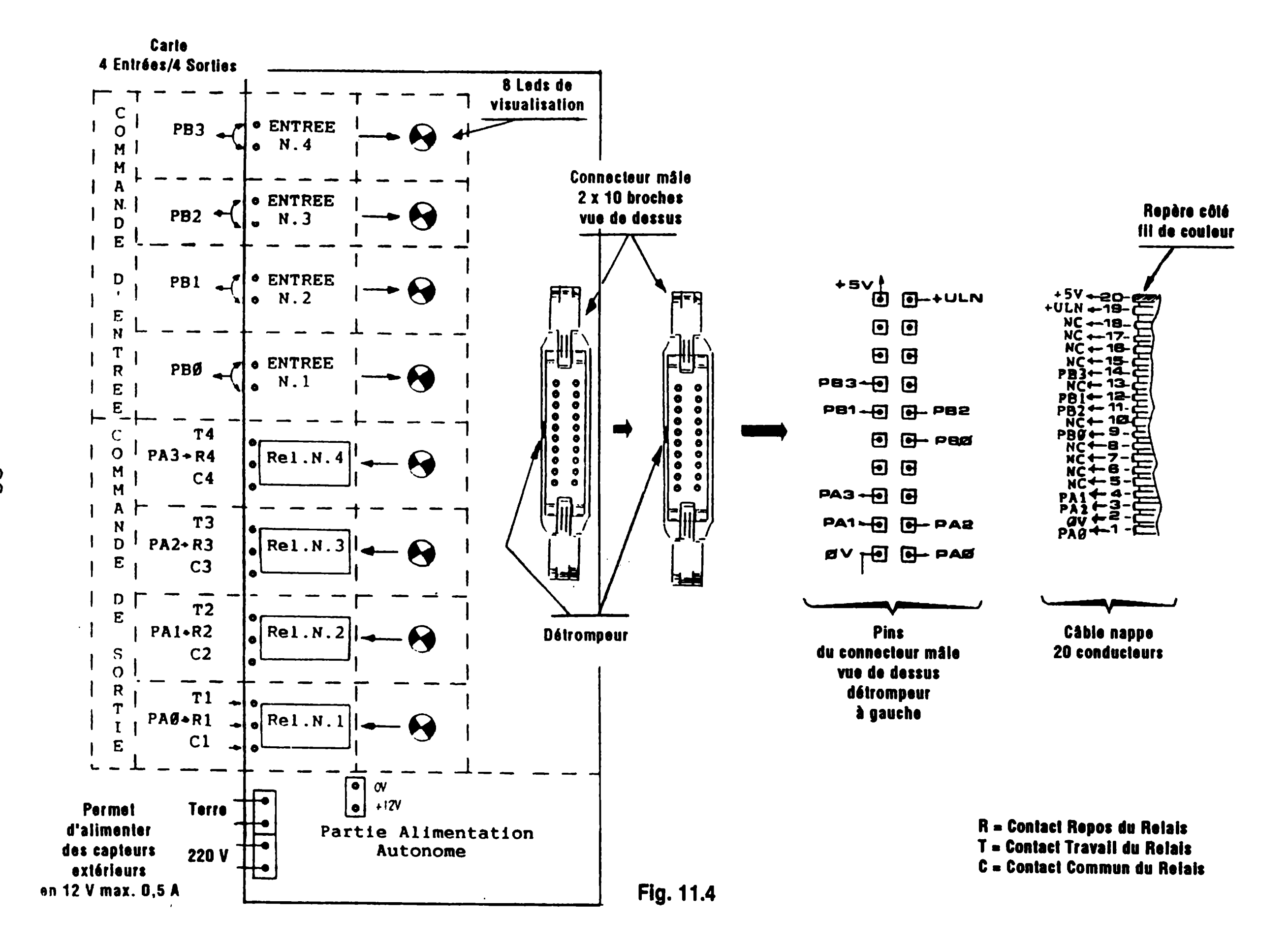

Carte
4 Entrées/4 Sorties
8 Leds de visualisation
Connecteur mâle 2 x 10 broches vue de dessus
Détrompeur
COMMANDE D'ENTREE
COMMANDE DE SORTIE
PB3
PB2
PB1
PBØ
ENTREE N.4
ENTREE N.3
ENTREE N.2
ENTREE N.1
T4 PA3+R4 C4
T3 PA2+R3 C3
T2 PA1+R2 C2
T1 PAØ+R1 C1
Rel.N.4
Rel.N.3
Rel.N.2
Rel.N.1
0V
+12V
Partie Alimentation Autonome
Terre
220 V
Permet d'alimenter des capteurs extérieurs en 12 V max. 0,5 A
+5V
+ULN
PB3
PB1
PB2
PBØ
PA3
PA1
PA2
ØV
PAØ
Pins du connecteur mâle vue de dessus détrompeur à gauche
+5V 20
+ULN 19
NC 18
NC 17
NC 16
NC 15
PB3 14
NC 13
PB1 12
PB2 11
NC 10
PBØ 9
NC 8
NC 7
NC 6
NC 5
PA1 4
PA2 3
ØV 2
PAØ 1
Repère côté fil de couleur
Câble nappe 20 conducteurs
R = Contact Repos du Relais
T = Contact Travail du Relais
C = Contact Commun du Relais

Fig. 11.4

CARACTÉRISTIQUES

Alimentation :

tension secteur : 220 V/50 Hz,
puissance maximale : 10 VA,
protection par fusible : 0,2 A,
tension max. des circuits d'entrées : + ou – 40 V,
courant max. (contact relais) des circuits de sortie : 10 A/250 V,
consommation maximale sur le + 5 V (alim. PC) : 90 mA,
résistance d'isolement : 10^{11} Ω,
tension d'isolement : 2 500 V.

CONNECTIQUE ET BRANCHEMENT (*fig.* 11.4)

Le 220 V sera à connecter entre les points marqués 220 V, la terre étant reliée à l'un des points du deuxième bornier.

CONNECTEUR DE BRANCHEMENT À LA CARTE PIA/PC, 16 ENTRÉES/SORTIES (*fig.* 11.5)

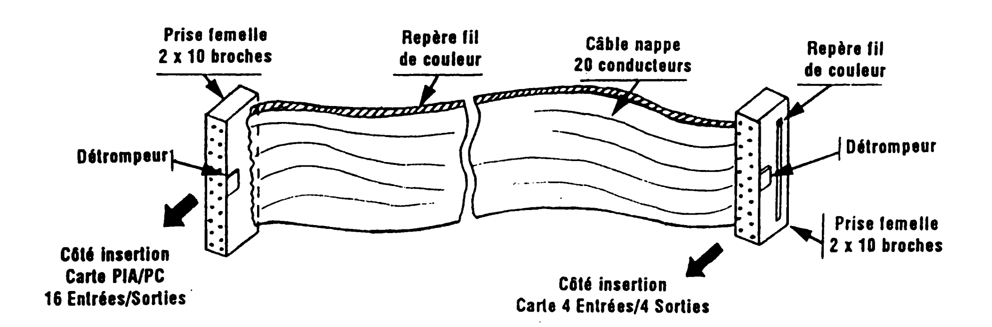


Fig. 11.5

TABLEAUX DES ÉTATS

Numéro des entrées	Condition des entrées	État des voyants	Niveau logique du port d'entrée 1 : Haut = + 5 V 0 : Bas = 0 V	Valeur BCD reçue
N° 1	Boucle de crt ouverte	Éteint	PB0 = 0	0
	Boucle de crt fermée	Allumé	PB0 = 1	1
N° 2	Boucle de crt ouverte	Éteint	PB1 = 0	0
	Boucle de crt fermée	Allumé	PB1 = 1	2
N° 3	Boucle de crt ouverte	Éteint	PB2 = 0	0
	Boucle de crt fermée	Allumé	PB2 = 1	4
N° 4	Boucle de crt ouverte	Éteint	PB3 = 0	0
	Boucle de crt fermée	Allumé	PB3 = 1	8

Numéro des sorties (ou relais)	Valeur BCD envoyée	Niveau logique du port de sortie.	État des voyants	Contact fermé des relais
N° 1	0	PA0 = 0	Éteint	C_1/R_1
	1	PA0 = 1	Allumé	C_1/T_1
N° 2	0	PA1 = 0	Éteint	C_2/R_2
	2	PA1 = 1	Allumé	C_2/T_2
N° 3	0	PA2 = 0	Éteint	C_3/R_3
	4	PA2 = 1	Allumé	C_3/T_3
N° 4	0	PA3 = 0	Éteint	C_4/R_4
	8	PA3 = 1	Allumé	C_4/T_4

CIRCUIT IMPRIMÉ

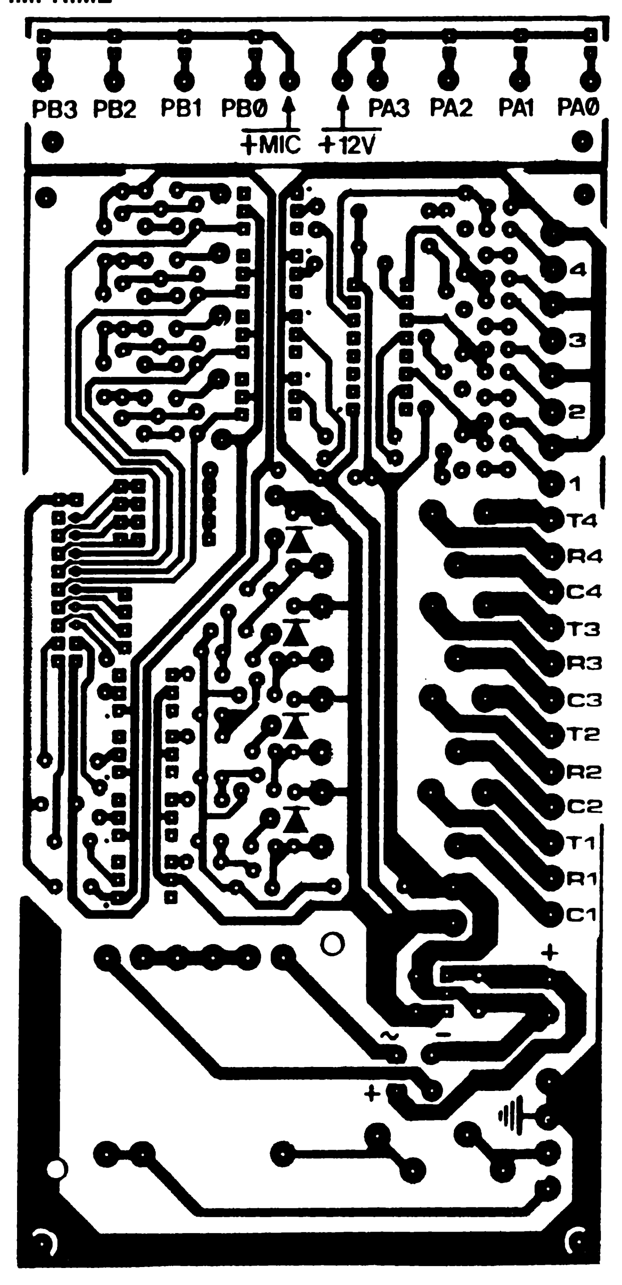


Fig. 11.6

MONTAGE (*fig.* 11.7)

Souder dans l'ordre :

— les straps au nombre de six,

— les diodes dont le sens est repéré par une bague. Certaines sont implantées debout et sont dessinées en vue de dessus (*fig.* 11.8),

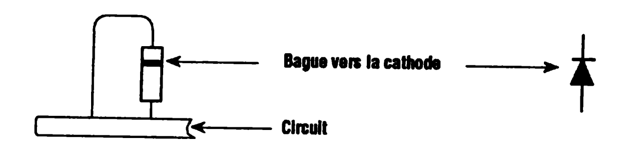


Fig. 11.8

— les résistances. Celles implantées debout sont dessinées

— le pont (attention au sens),

— les condensateurs : attention à la polarité du chimique,

— le régulateur 7812 (sens repéré par la partie métallique),

— le porte fusible (en deux parties),

— les 4 résistances ajustables,

— les transistors BC 337 25 dont le sens est repéré par le méplat,

— les huit optocoupleurs MCT 2E : attention au sens indiqué par le point,

— le 4049 repéré par un point ou une encoche,

— les treize borniers à vis,

— le connecteur mâle 2 × 10 broches,

— les quatre relais,

— le transformateur,

— les huit LED sur le circuit visualisation en respectant la polarité : patte longue au plus.

Câbler le circuit visualisation sur le circuit principal à l'aide de fil isolé et en reliant entre eux les points marqués de la même manière.

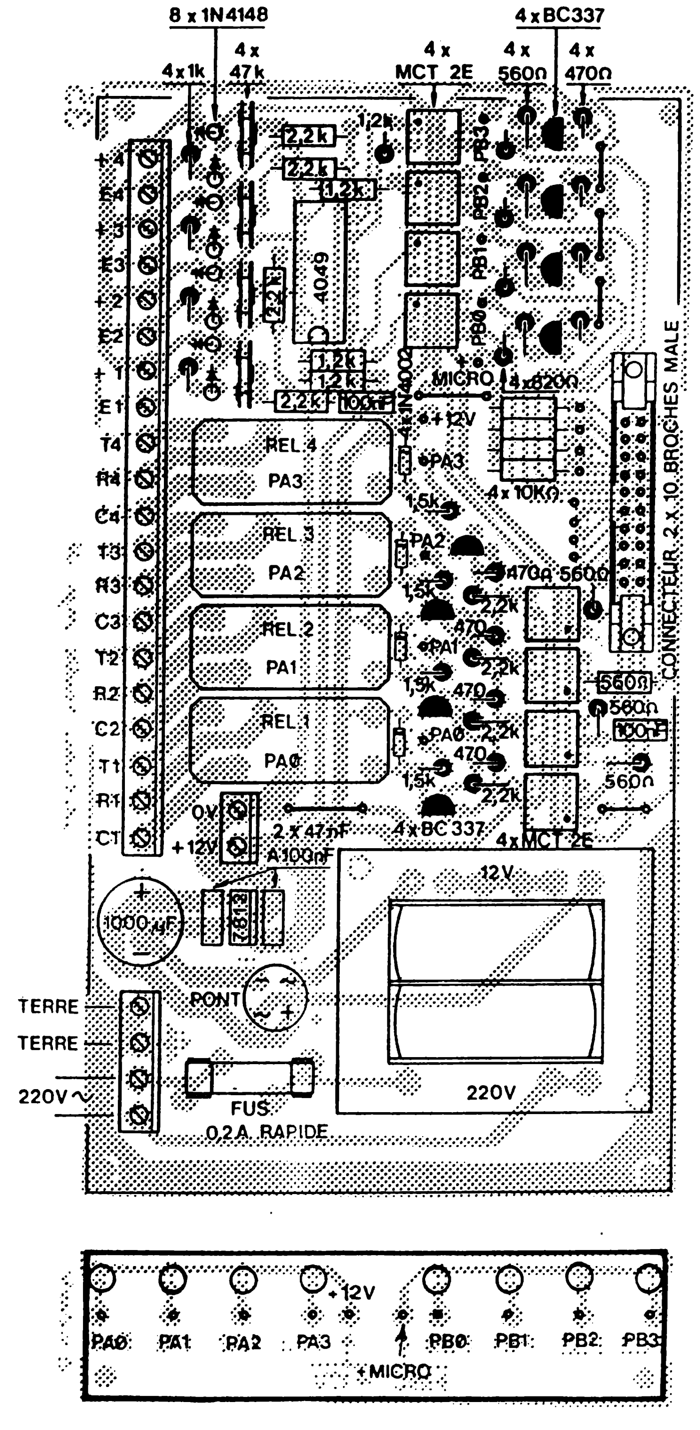
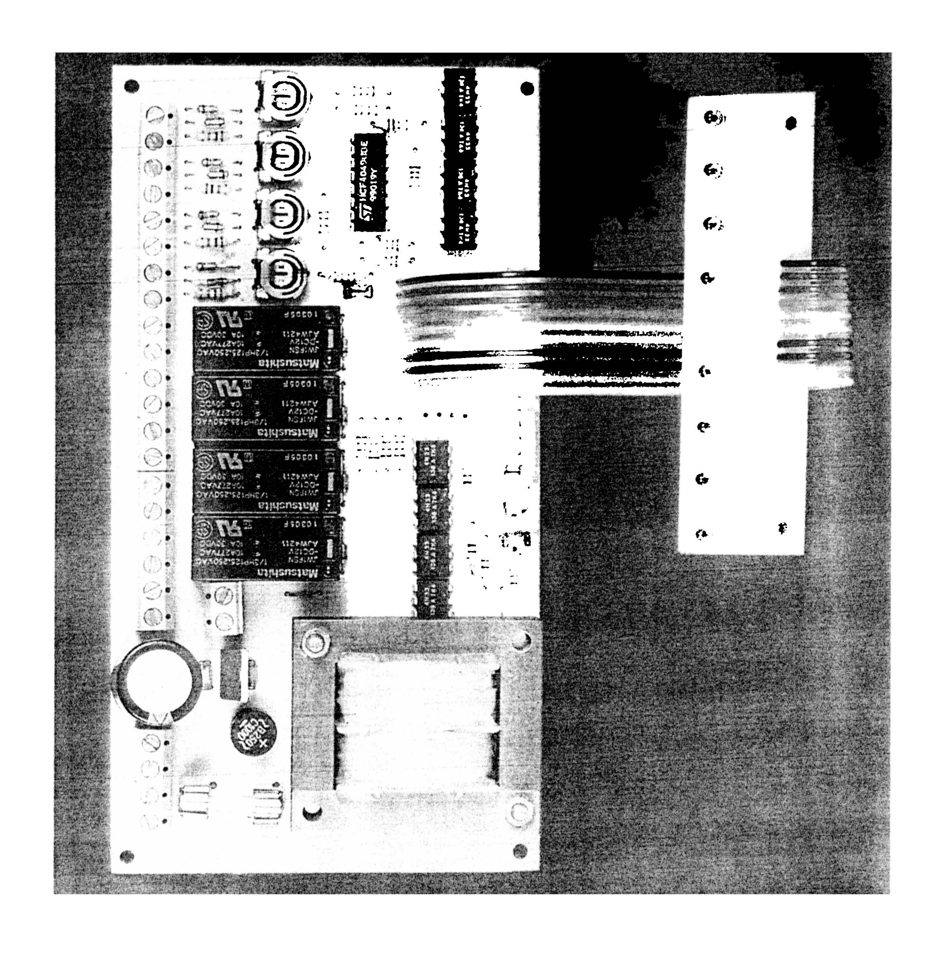


Fig. 11.7

8 LED Ø 3



Carte principale		*Carte visualisation*
PB3	avec	PB3
PB2	avec	PB2
PB1	avec	PB1
PB0	avec	PB0
+ micro	avec	+ micro
+ 12 V	avec	+ 12 V
PA3	avec	PA3
PA2	avec	PA2
PA1	avec	PA1
PA0	avec	PA0

LISTE DES COMPOSANTS

huit optocoupleurs MCT 2E,
1 × 4049,
8 × BC 337-25,
4 × 1N 4002,
8 × 1N 4148,
8 × LED 3 mm,
13 × borniers à vis pour CI,
8 × 470 Ω,
8 × 560 Ω,
4 × 820 Ω,
4 × 1 kΩ,
4 × 1,2 kΩ,
4 × 1,5 kΩ,
8 × 2,2 kΩ,
4 × 10 kΩ,
un connecteur informatique mâle 2 × 10 broches,
quatre ajustables verticales : 47 kΩ,
circuit imprimé percé,
quatre relais 12 V/1RT/10 A.

LISTE DES COMPOSANTS PARTIE ALIM. AUTONOME

transformateur 12 V/10 VA,
deux supports fusible pour CI,
fusible 0,2 A rapide,
pont 1,5 A,
1 000 µF/25 V radial,
4 × 47 à 100 nF,
régulateur 7812.

TEST DE VOTRE CARTE 4 ENTRÉES/4 SORTIES

Réunir les points : E_4 à T_4 + 4 à C_4; E_3 à T_3 + 3 à C_3; E_2 à T_2 + 2 à C_2; E_1 à T_1 + 1 à C_1, c'est-à-dire entrée 4 vers sortie 4, etc.

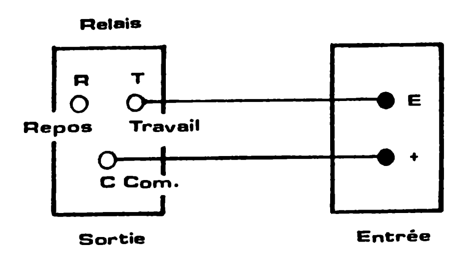


Fig. 11.9

Sous MS DOS, à l'aide de la disquette, charger le programme ROBOT PC. Lorsque le menu apparaît, sélectionner le programme 1 en composant 1 puis entrée (*enter*) (Programme test entrée sortie).

L'écran « programme Carte entrée sortie » apparaît.

Composer 1 puis entrée (*enter*), le relais doit coller et sur l'écran doit apparaître entrée 1 active=1. Valeur entrée:1.

En composant 9, puis entrée (enter), les relais 1 et 4 doivent coller. On a alors entrée 1 et 4 actives avec leur valeur binaire. La valeur transmise à l'ordinateur est 9.

En composant 15 (1+2+4+8), les quatre relais collent. L'ordinateur reçoit à travers les entrées une valeur 15.

PROGRAMME

Le programme ENTSORT est disponible sur la disquette en basic (BASICA ou GW Basic de Microsoft).

Après avoir chargé le basic, composez RUN "ENTSORT".

Ce programme permet de commander les relais en affichant la valeur de la commande puis de tester les entrées de la carte.

Si on réunit les entrées de la carte aux sorties (entre C et T) des relais, le nombre transmis à l'ordinateur doit correspondre au nombre composé.

Exemple :

— Si vous composez 14, la commande des relais de poids binaire respectivement 8, 4 et 2, qui correspond aux relais 2, 3 et 4. La valeur transmise à l'ordinateur doit être aussi de 14.

— Si vous composez 255, tous les relais doivent coller, mais la valeur transmise à l'ordinateur ne sera que de 15. En effet, le programme peut commander jusqu'à huit relais (poids binaire d'un octet, soit 128, 64, 32, 16, 8, 4, 2 et 1). Cependant, la carte ne comportant que quatre entrées, l'ordinateur recevra la somme des quatre relais, soit 1, 2, 4, 8 = 15. Les quatre autres commandes 128, 64, 32 et 16 n'étant pas utilisées.

Relais 1 commande = 1,
Relais 2 commande = 2,
Relais 3 commande = 4,
Relais 4 commande = 8;

Entrée 1 active valeur transmise = 1,
Entrée 2 active valeur transmise = 2,
Entrée 3 active valeur transmise = 4,
Entrée 4 active valeur transmise = 8.

```
10 '--programme---ENTSORT---
20 CLS
30 KEY OFF
40 COLOR Ø,7:LOCATE ,22:PRINT " Test carte entrées sorties en basic "
50 'config pia-------------
60 P=&H313
70 O=&H31Ø
80 R=&H311
90 OUT P,&H82
1ØØ '---commande--des--sorties--
110 LOCATE 25,26:PRINT" RETOUR au BASIC composer -1 ";:COLOR 7,Ø
120 LOCATE 12,30 :PRINT "Valeur sortie=        "
130 LOCATE 12,47 :INPUT A
140 IF A=-1 THEN CLS:END
15Ø LOCATE 12,49:COLOR Ø,7:PRINT "█████ ";A;" █"
160 COLOR 7,Ø
170 IF A>255 OR A<-1 THEN BEEP:GOTO 120
180 OUT O,A
19Ø FOR X=1 TO 1ØØ0:NEXT X
2ØØ '----test--de--la--valeur--d'entrée--
21Ø B=INP(R)
22Ø LOCATE 14,22:PRINT "Valeur transmise à l'ordinateur=
23Ø LOCATE 14,56:PRINT B
24Ø GOTO 120
```

EXPLICATION DU PROGRAMME

ligne 20 : RAZ écran,

ligne 30 : éteindre ligne 25 de l'écran,

ligne 60 : P = adresse configuration du 8255.

ligne 70 : O = adresse port en sortie,

ligne 80 : R = adresse port en entrée,

ligne 90 : transfert dans P de 82 qui configure adresse 0 en sortie et adresse de R port en entrée,

ligne 130 : entrée de la valeur à transmettre en sortie, d'où A,

ligne 140 : si valeur = – 1 RAZ écran, retour en Basic,

ligne 150 : impression de la valeur de sortie,

ligne 170 : si la valeur est inférieure à – 1 ou supérieure à 255 retour à entrée de la valeur,

ligne 180 : valeur transmise à la carte 4 entrées/4 sorties,

ligne 190 : temporisation pour laisser aux relais le temps de coller,

ligne 210 : transfert dans B de la valeur présente sur les entrées de la carte entrées/sorties,

ligne 220 : impression de la valeur,

ligne 230 :

ligne 240 : retour à la saisie de la commande des sorties.

RACCORDEMENT ENTRÉES ET SORTIES

CARTE 4 ENTRÉES/4 SORTIES

Les entrées

Chacune des entrées est raccordée à un bornier deux plots

Entrée 1 { E_1 → signal entrée,
+ 1 → + 12 V carte 4 E/ES;

Entrée 2 { E_2 → signal entrée,
+ 2 → + 12 V;

Entrée 3 { E_3 → signal entrée,
+ 3 → + 12 V;

Entrée 4 { E_4 → signal entrée,
+ 4 → + 12 V.

Schéma (*fig.* 11.10)

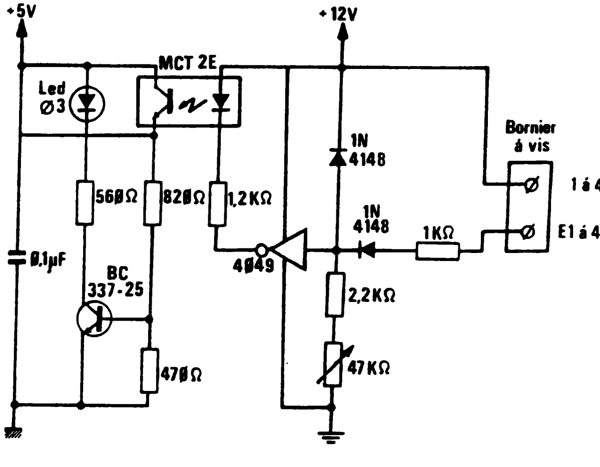


Fig. 11.10

Sur chacune des entrées le plus est actif sur l'entrée E. Les raccordements suivants sont possibles (*fig.* 11.11).

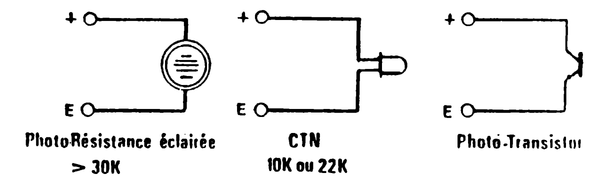


Fig. 11.11

Sur ces trois montages (*fig.* 11.12), le seuil de basculement est réglable grâce à la résistance ajustable 47 kΩ sur chacune des entrées.

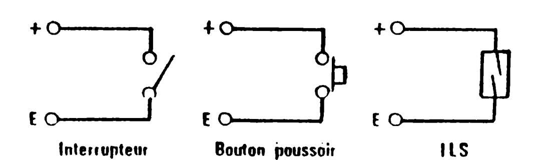
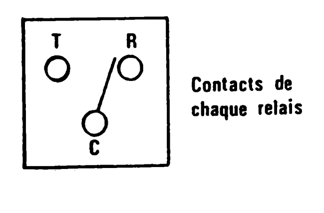


Fig. 11.11

Les sorties

Chaque relais a son contact travail, repos et commun reliés à un point des borniers (*fig.* 11.13).

Chaque contact supporte jusqu'à 10 ampères.



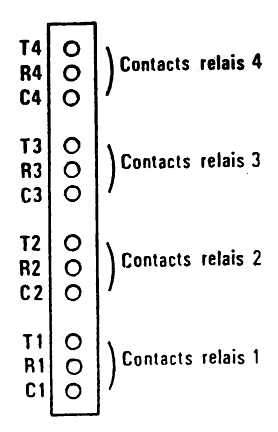


Fig. 11.13

Bornier pour alimentation de capteur extérieur

Ce bornier permet d'alimenter en 12 V (0,5 A au maximum) un ou des capteurs extérieurs (*exemple :* barrière infrarouge, temporisateur, etc.) (*fig.* 11.14).

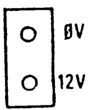


Fig. 11.14

⑫ Clavier numérique

TECHNOLOGIE

Le périphérique «clavier» permet de décoder onze touches numérotées de 0 à 10 et d'envoyer un bip sonore signifiant qu'il y a une action sur une touche. Pour ce faire, le clavier utilise un encodeur de priorité 40147 ainsi qu'un avertisseur sonore constitué avec un circuit intégré 4069 et un transducteur piézo.

CONNECTION

Le périphérique *clavier numérique* est connectable à l'interface PIA/PC, vingt-quatre entrées/sorties (disponible en kit).

Le clavier est alimenté directement par le 5 V du PC (on peut éventuellement l'alimenter extérieurement en cas d'utilisation sans le PC).

SCHÉMA DE PRINCIPE (*fig.* 12.1)

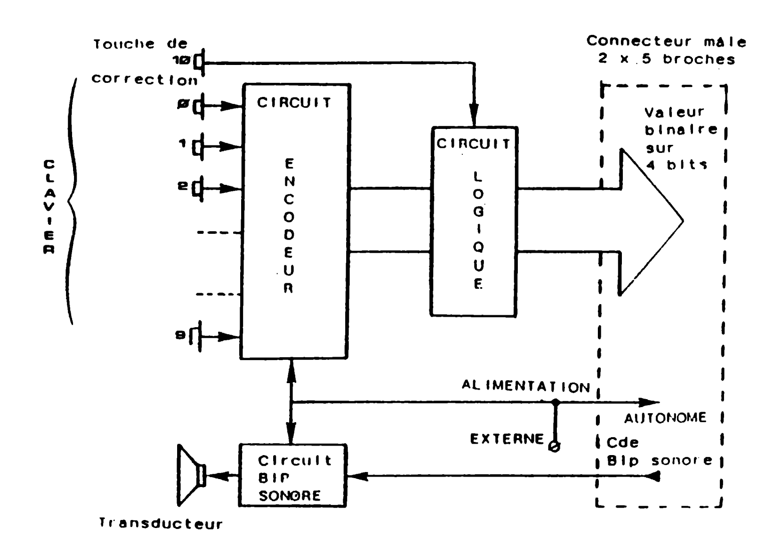


Fig. 12.1

CIRCUIT ENCODEUR

Le circuit CMOS 40147 effectue un codage des dix entrées en un mot BCD (binaire codé décimal) sur quatre bits.

Si aucune touche n'est actionnée, les dix entrées numérotées de 0 à 9 sont au niveau logique haut grâce aux réseaux de résistances reliés au + 5 V.

L'action d'une touche entraîne un niveau logique bas sur l'entrée concernée.

Mot BCD / Touche actionnée	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$	Valeur BCD	Valeur complémentaire du mot BCD
	D	C	B	A		
Aucune touche actionnée	0	0	0	0	0	15
Touche : 0	1	1	1	1	15	0
Touche : 1	1	1	1	0	14	1
Touche : 2	1	1	0	1	13	2
Touche : 3	1	1	0	0	12	3
Touche : 4	1	0	1	1	11	4
Touche : 5	1	0	1	0	10	5
Touche : 6	1	0	0	1	9	6
Touche : 7	1	0	0	0	8	7
Touche : 8	0	1	1	1	7	8
Touche : 9	0	1	1	0	6	9

Remarque

On constate que les bits A, B, C, D sont inversés, cela est dû à la structure interne du circuit intégré 40147. Pour obtenir la valeur BCD correcte, il suffit de prendre le complément du mot. Ceci est facilement réalisable par logiciel.

TOUCHE 10

Elle est constituée de diodes et de résistances, elle ajoute un mot binaire, quand on actionne la touche correction (ou n.10), différent des autres mots binaires déjà énumérés dans le tableau précédent.

Remarque

En prenant le complémentaire du mot binaire, on obtient la valeur BCD:10.

Mot BCD / Touche actionnée	D	C	B	A	Valeur BCD	Valeur complémentaire
Touche correction du 10	0	1	0	1	5	10

CIRCUIT BIP SONORE

Il permet de matérialiser la prise en compte d'un mot binaire à l'utilisateur, par un bip sonore.

Le bip sonore est réalisé par un oscillateur, qui retentit seulement si l'entrée PC4 est au niveau logique haut.

CARACTÉRISTIQUES

— Tension d'alimentation de 4,5 à 6 V continue.

— Consommation au repos (sans action sur les touches) de 3 µA.

— Consommation maximale de 5 mA. Clavier onze touches.

— Connecteur informatique 2 × 5 broches.

SCHÉMA DU CLAVIER NUMÉRIQUE

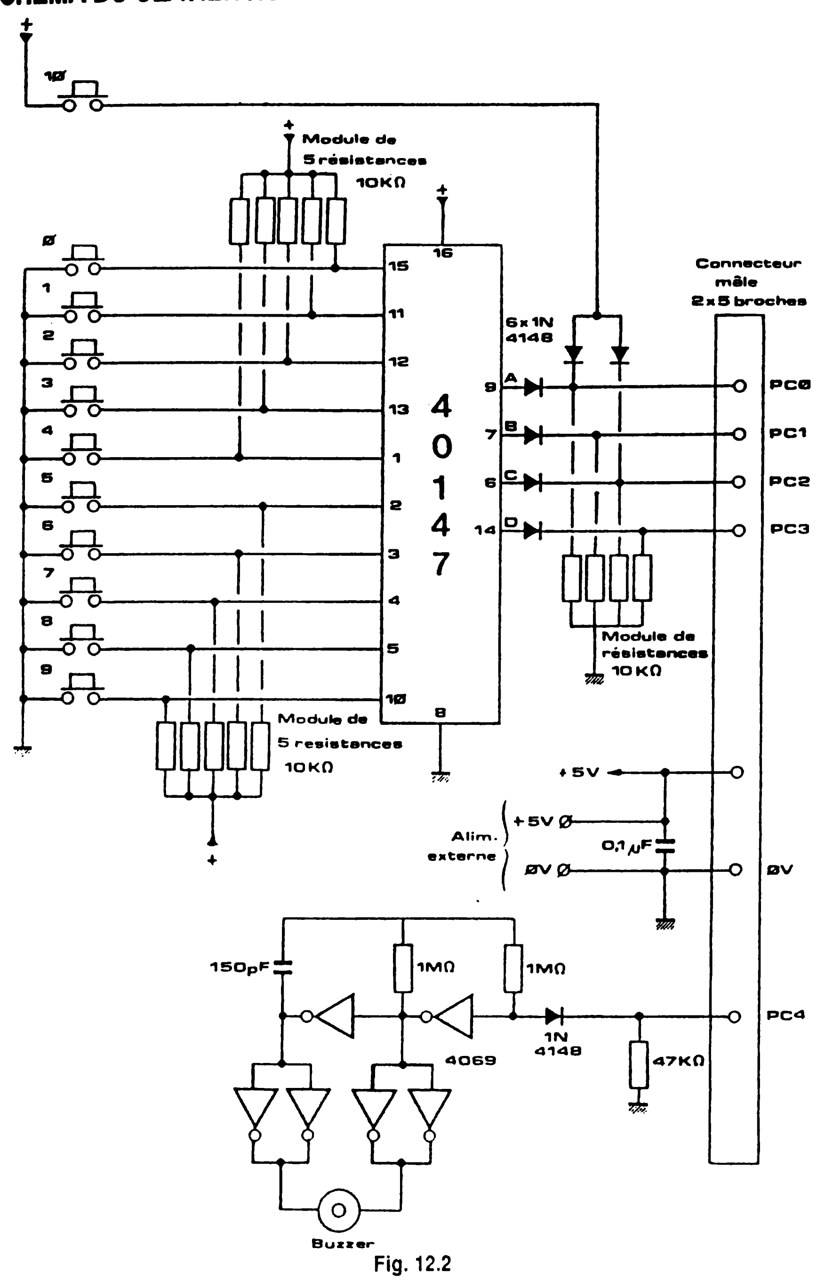


Fig. 12.2

EXEMPLE DE PROGRAMMATION DU CLAVIER EN BASIC (GW BASIC DE MICROSOFT)

Ce programme simple permet de visualiser à l'écran le numéro de la touche actionnée, et également d'activer le bip sonore lors du relâchement de celle-ci.

PROGRAMME

```
10  OUT &H313, &H83
20  CLS
30  V=INP (&H312): IF V=0 GOTO 30
40  PRINT "Numéro de la touche actionnée:";:PRINT (15-V)
50  IF INP (&H312)=V GOTO 50
60  OUT &H312,16
70  FOR X=1 TO 300: NEXT X
80  OUT &H312,0: GOTO 20
```

EXPLICATIONS DU PROGRAMME

ligne 10 : initialisation de la carte PIA/PC demi port C inférieur (PC0... PC3) en ENTREE, demi port C supérieur (PC4...PC7) en SORTIE (voir notice carte PIA/PC),

ligne 20 : effacement de l'écran,

ligne 30 : chargement dans V du mot binaire: tant qu'une touche n'est pas actionnée, le mot binaire est égal à zéro, donc le programme reste en ligne 30 jusqu'à l'action d'une touche,

ligne 40 : affichage à l'écran du numéro de la touche actionnée. (15-V) permet d'afficher le complément du mot binaire afin d'obtenir la valeur BCD correcte,

ligne 50 : tant que la touche est enfoncée, on reste à la ligne 50. Dès qu'elle est relâchée, on va à la ligne suivante. Ce procédé évite de prendre plusieurs fois le même mot binaire pendant l'activation d'une touche,

ligne 60 : on charge le demi port C supérieur logique haut. Le circuit bip sonore est activé,

ligne 70 : temporisation nécessaire à l'établissement du bip sonore. Cette temporisation devra être augmentée ou diminuée selon la fréquence d'horloge de votre PC,

ligne 80 : arrêt du bip sonore, puis retour au début du programme ligne 20,

PORTS ENTRÉES/SORTIES DU CLAVIER NUMÉRIQUE (*fig.* 12.3)

Port de sorties : repéré sur le schéma de la figure 12.2 par PC0 à PC3, il est raccordé au port d'entrées de la carte PIA/PC. Un niveau logique haut sur l'un de ces bits signifie une tension d'environ + 5 V; un niveau logique bas signifie une tension nulle.

Port d'entrée : seul un bit du port PC4 (*fig.* 12.2) est raccordé au port de sortie de la carte PIA/PC.

Il y a activation du bip sonore si PC4 se trouve au + 5 V (niveau logique haut). Il y a désactivation si son entrée est au niveau logique bas (0 V) ou en l'air.

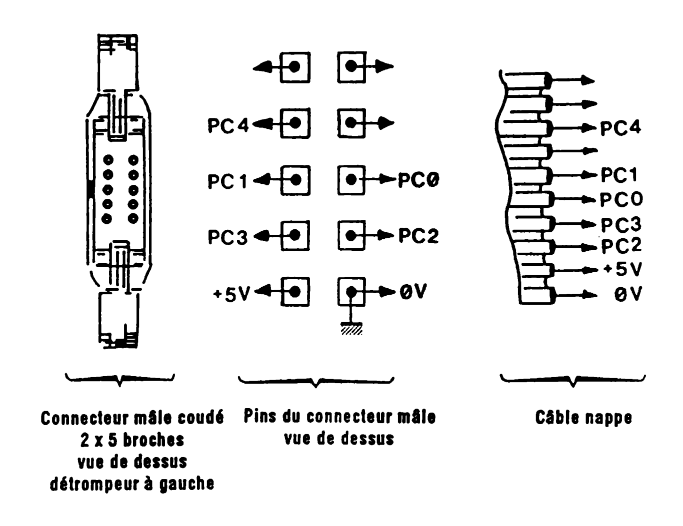
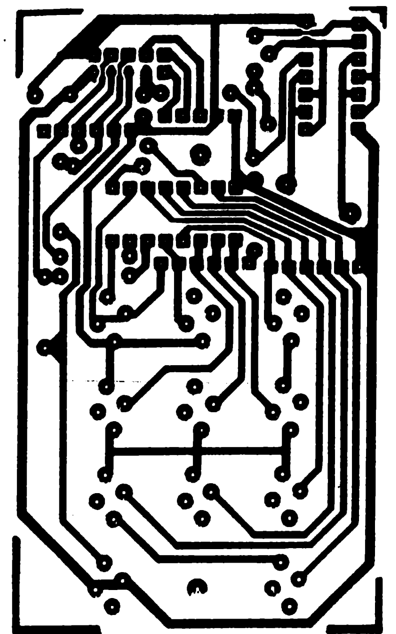


Fig. 12.3



CIRCUIT IMPRIMÉ ET IMPLANTATION

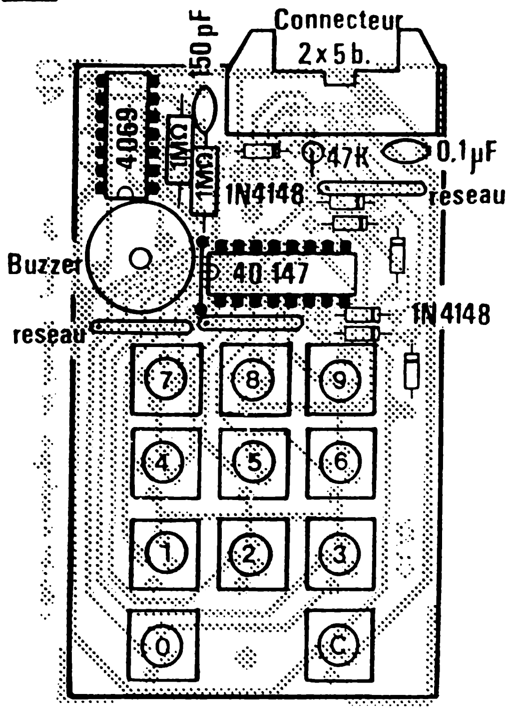
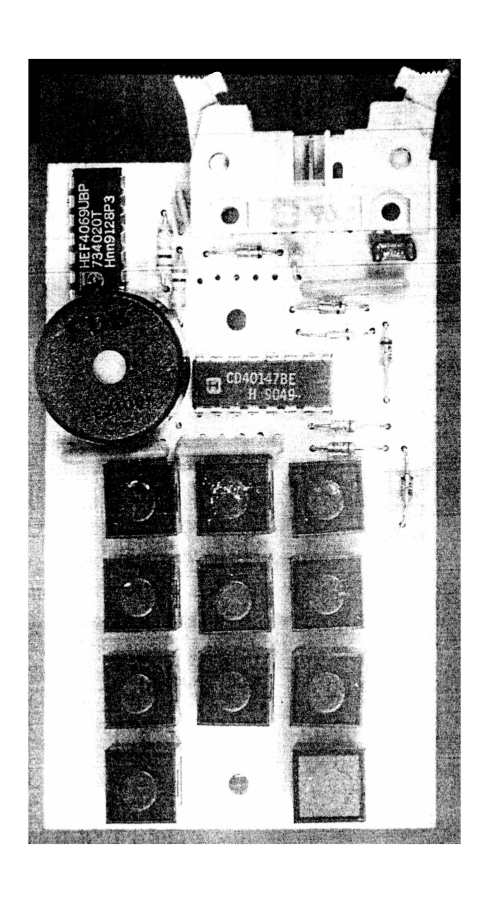


Fig. 12.4


HEF4069UBP
734020T
Hnn9128P3
CD40147BE
H 9049

LISTE DES COMPOSANTS

47 kΩ (jaune violet orange),
2 × 1 MΩ (marron noir vert),
trois réseaux de résistances 706A103,
1 × 150 pF,
1 × 0,1 µF,
7 × 1N 4148,
1 × 4069,
1 × 40147,
un buzzer piézo,
Connecteur mâle coudé 2 × 5 broches,
dix touches CI noires,
une touche CI rouge,
un circuit imprimé percé.

MONTAGE

Souder dans l'ordre et le plus près possible du circuit imprimé :

— le strap repéré •—• sur l'implantation;

— les résistances. La 47 kΩ est à souder debout et est représentée sur l'implantation ⊖— ;

— les diodes dont le sens est donné par une bague —[▯]—;

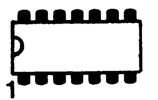
— les condensateurs;

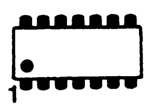
— les trois réseaux de résistances ▱ ; attention au point;

— les onze touches, la rouge étant celle repérée C (correction);

— le buzzer (sens indifférent);

— les circuits intégrés repérés par un point ou une encoche :





— le connecteur 2 × 5 broches.

PROGRAMME "PCLAV"

Sous GW BASIC ou BASICA de Microsoft.

Chaque fois que l'on appuie sur une touche du clavier, la valeur de celle-ci est affichée.

Pour sortir du programme : CTRL + Pause (ou Breack).

```
5 '----programme---PCLAV---
10        REGCON = &H313
20        PORTC  = &H312
30        OUT REGCON,&H81
40        KEY OFF
50        CLS
60        CLAVIER=0
70        LOCATE 25,20:COLOR 0,7
80        PRINT" Test clavier mini PC :appuyer sur une touche. ";
90        COLOR 7,0
100       LOCATE 10,22
110       V=(INP(PORTC)AND 15)
120       IF V=0 GOTO 110
130       OUT PORTC,16
140       FOR I=1 TO 150:NEXT I
150       OUT PORTC,0
160       CLAVIER=15-V
170       IF (INP(PORTC)AND 15) =V GOTO 170
180       PRINT"touche appuyée="; CLAVIER;:
190       IF (CLAVIER=10) THEN PRINT" Touche Correction.":GOTO 100
200       PRINT SPC(30):GOTO 100
```

EXPLICATION DU PROGRAMME PCLAV

ligne 10 : configuration du PIA registre de configuration en H313,

ligne 20 : Port C en H312,

ligne 30 : configuration de C, 4 bits en entrée, 4 bits en sortie (demi port haut en sortie, demi port bas en entrée),

ligne 40 : éteindre ligne 25 de l'écran,

ligne 50 : RAZ écran,

ligne 60 : vidage registre clavier,

ligne 70 : affichage inversé. Test clavier,

ligne 80 :

ligne 90 : retour affichage normal,

ligne 100 : ligne 10 en 22, position (préparation de l'affichage),

ligne 110 : entrée du demi port C dans registre V,

ligne 120 : si V = 0 retour à 110 (pas de touche appuyée),

ligne 130 : sortie demi port C avec valeur 16 (commande du bip sur mini clavier),

ligne 140 : temporisation,

ligne 150 : arrêt bip sonore (sortie de la valeur 0),

ligne 160 : transfert de V complémenté à 15 dans registre clavier,

ligne 170 : si la valeur à l'entrée de l'ordinateur est toujours = à la valeur entrée précédemment, on reste en 170 (touche pas relâchée),

ligne 180 : affichage "touche appuyée = avec valeur de la touche,

ligne 190 : si clavier = 10 (touche appuyée = touche rouge) affichage touche correction. Retour à 100,

ligne 200 : affichage 30 espaces (effacement ligne précédente) retour à 100.

PROGRAMME "PCLAV2"

Sous GW BASIC ou BASICA de Microsoft.

```
5 '--programme--PCLAV2--
10 CLS
20 REGCON = &H313
30 PORTC  = &H312
40 OUT REGCON,&H81
50 KEY OFF
60 LOCATE 10,22:INPUT "code secret";SECRET
70 LOCATE,22:PRINT SPC(40)
80 CLS:COLOR 7,0
90 CLAVIER=0
100 LOCATE 25,20:PRINT" Test clavier mini PC :appuyer sur une touche. ";
110 LOCATE 10,22 :CLAV=0
120 V=(INP(PORTC)AND 15)
130 IF V=0 GOTO 120
140 IF CLAVIER=10 THEN PRINT SPC(50):LOCATE 10,22
150 OUT PORTC,16
160 FOR I=1 TO 150:NEXT I
170 OUT PORTC,0
180 CLAVIER=15-V
190 IF (INP(PORTC)AND 15) =V GOTO 190
200 IF (CLAVIER=10) THEN LOCATE 10,22:PRINT" entrée= ";CLAV;"      ":GOTO 240
210 CLAV=CLAV*10+CLAVIER
220 PRINT USING"#";CLAVIER;
230 GOTO 120
240 COLOR 16,7:LOCATE 10,44
250 IF CLAV=SECRET THEN PRINT " C'est le bon code "ELSE PRINT   "       Erreur
260 COLOR 7,0:GOTO 100
```

Le programme vous permet d'entrer un code secret au clavier numérique de l'ordinateur.

Composer ce code à l'aide du mini clavier, puis appuyer sur la touche rouge qui fait office de touche entrée (enter). L'ordinateur doit vous afficher "c'est le bon code". Recomposer un code faux au mini clavier, puis touche rouge. L'ordinateur doit vous afficher "ERREUR".

Pour sortir du programme, appuyer sur CTRL et Pause (ou Breack) en même temps.

EXPLICATION DU PROGRAMME PCLAV2

lignes 20 à 40 : configuration de la carte PIA,

ligne 50 : entrée sur le clavier de l'ordinateur du code secret,

ligne 120 : entrée de la valeur de la touche appuyée sur mini clavier,

ligne 130 : si valeur = 0 retour à 120 (pas de touche appuyée),

ligne 140 : si la valeur entrée la fois précédente = 10 (touche rouge) effacement de la ligne affichée,

ligne 150 : commande du bip sonore sur mini clavier,

ligne 160 : temporisation,

ligne 170 : arrêt du bip sonore,

ligne 180 : complément à 15 de la valeur entrée dans clavier,

ligne 190 : si la valeur à l'entrée est toujours la même on reste en 190 (test si touche relâchée),

ligne 200 : si entrée = 10 (touche rouge) affichage de "entrée=" Clav puis aller en 240,

ligne 210 : multiplication du chiffre entrée la fois précédente par 10 et addition au chiffre qui vient d'être entrée avec résultat dans CLAV,

ligne 220 : impression du dernier chiffre entré,

ligne 230 : retour à 120 (entrée de la valeur de la touche),

ligne 240 : couleur clignotante,

ligne 250 : si le nombre composé = le nombre secret, affichage "c'est le bon code", sinon, affichage "erreur",

ligne 260 : couleur normale, retour à 100.

⑬ Chenillard PC

Le chenillard PC vous séduira par sa simplicité, pour animer vos soirées, donner un air de fête à vos vitrines ou bien faire défiler vos textes.

Aucune alimentation extérieure n'est à prévoir grâce à l'emploi d'opto-triac.

Votre PC est totalement protégé du secteur véhiculé par les triacs grâce à cet opto-isolation.

Des exemples simples en Basic (GW Basic, marque déposée de Microsoft) vous permettront d'écrire vos propres applications.

TECHNOLOGIE

L'interface chenillard est constituée autour des circuits intégrés MOC 3020 (optocoupleur) associés à des triacs.

Cette interface assure une parfaite isolation électrique entre l'ordinateur et le secteur.

CONNEXIONS

Le périphérique «chenillard» est connectable à l'interface PIA/PC (disponible en kit ou monté).

L'alimentation est raccordée directement au secteur, et permet d'alimenter jusqu'à huit ampoules (220 V/60 W).

SCHÉMA DE PRINCIPE

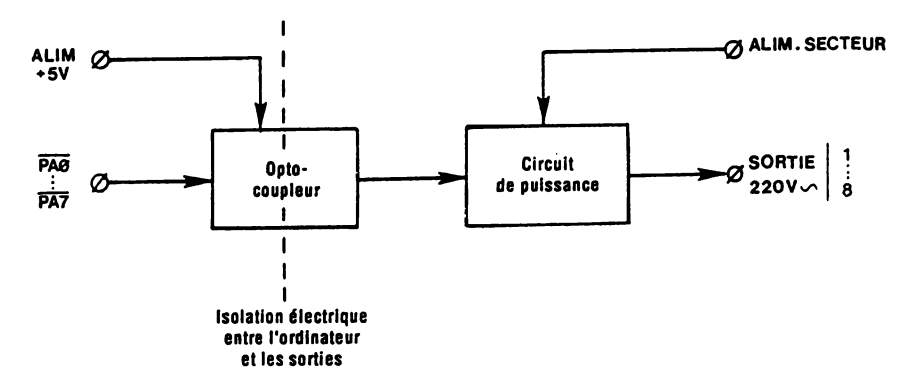


Fig. 13.1

DESCRIPTION DES ÉLÉMENTS

Optocoupleur

L'optocoupleur est réalisé par un optotriac qui permet un isolement galvanique de l'ordinateur et de la partie puissance du montage. Quand la diode LED de l'optocoupleur est passante ($\overline{PA0}$(PIA) = 1 entrée carte $\overline{PA0}$); elle émet pour débloquer le triac de l'optocoupleur qui commandera la gâchette du triac de la partie puissance.

Les résistances

Les résistances de 560 Ω sont câblées pour limiter les intensités dans la diode émettrice de l'optocoupleur et dans la gâchette du triac de la partie puissance.

Le circuit RC

Le circuit RC de la partie puissance permet d'éviter les montées brusques de tension à la commutation du triac.

SCHÉMA DE PRINCIPE

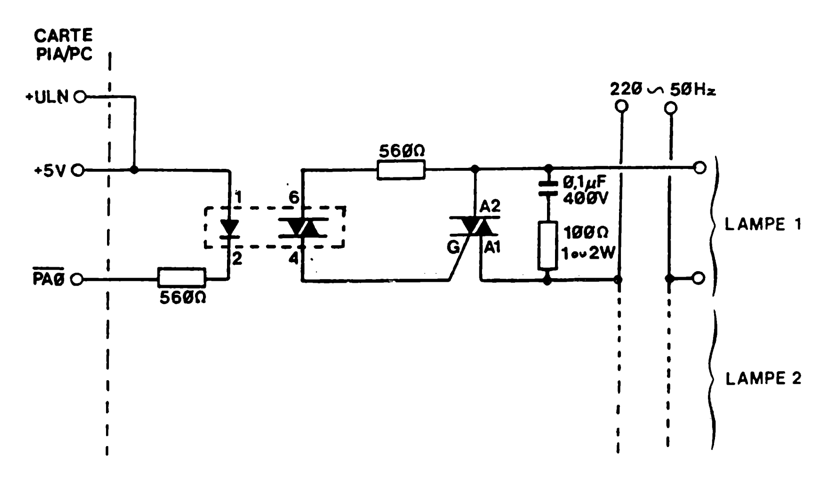


Fig. 13.2

MONTAGE (*fig.* 13.3 et 13.4)

Souder dans l'ordre suivant et le plus près possible du circuit :

— les sept straps,

— les résistances,

— les condensateurs,

— les optotriacs MOC 3020 dont le sens d'insertion est repéré par un point ou une encoche,

— les triacs repérés par le coin tronqué qui correspond à la gâchette,

— les neuf borniers à vis en orientant le côté sortie fil vers l'extérieur de la carte,

— le connecteur coudé 2 × 10 broches (*fig.* 13.5).

Câbler le fil secteur sur le bornier repéré 220 V.

Câbler les spots sur les huit borniers de sortie (maximum huit spots de 60 W).

Relier le chenillard PC à votre carte interface PC ou votre carte huit sorties à l'aide d'un câble informatique vingt conducteurs.

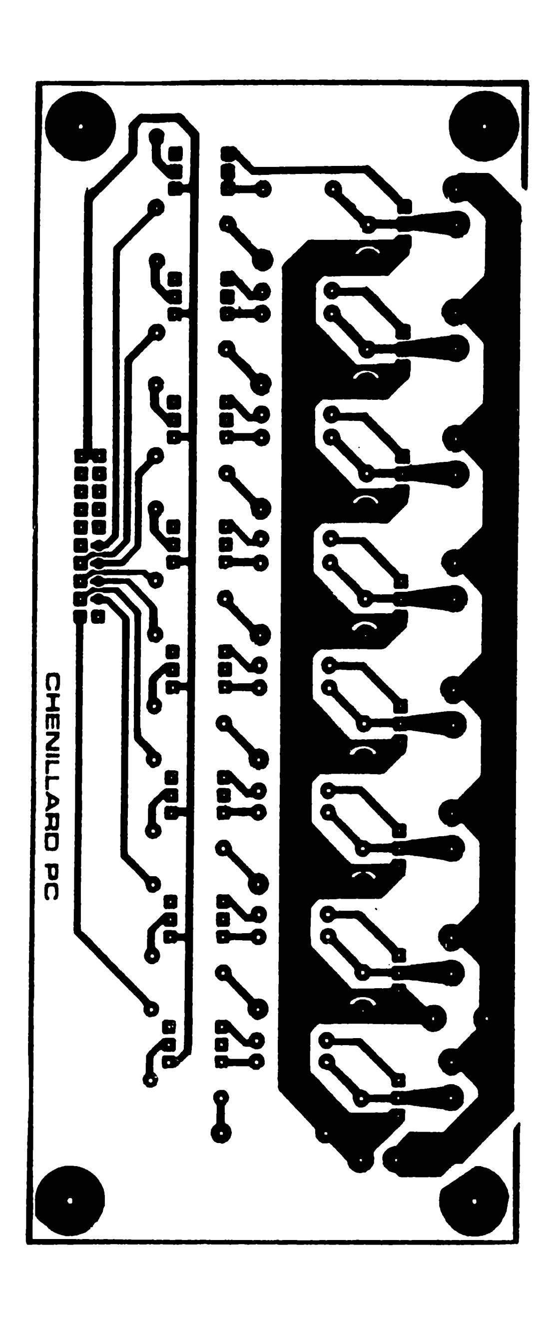


Fig. 13.3

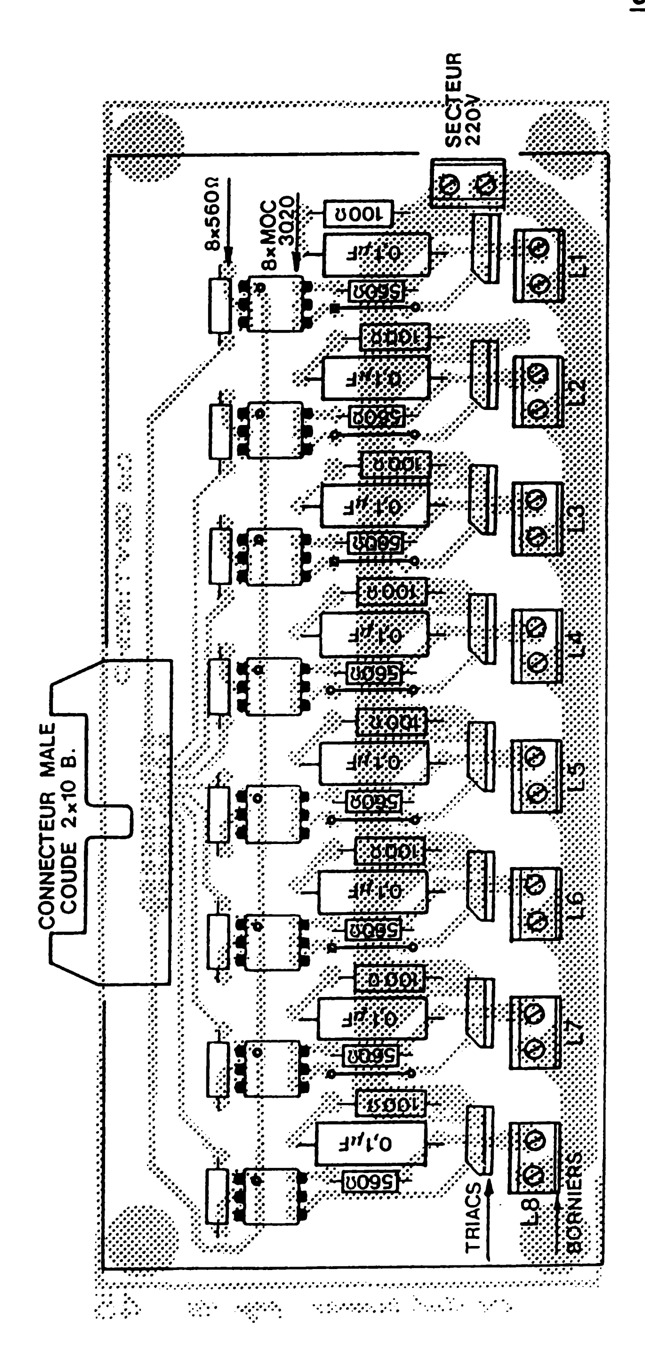

SECTEUR
220V
8×560Ω
8×MOC
3020
100Ω
0.1µF
560Ω
CONNECTEUR MALE
COUDE 2×10 B.
L1
L2
L3
L4
L5
L6
L7
L8
TRIACS
BORNIERS

Fig. 13.4

Connecteur 2 x 10 broches

Pins du connecteur mâle vue de dessus

Câble nappe

Fig. 13.5

LISTE DES COMPOSANTS

huit optocoupleurs MOC 3020,
huit triacs ZO 409,
seize résistances 560 Ω, 1/4 W 5 %,
huit résistances 100 Ω, 1 ou 2 W 5 %,
huit condensateurs 0,1 μF/400 V,
un connecteur informatique 2 x 10 broches mâle coudé,
neuf borniers à vis deux plots.

Bien vérifier la parfaite isolation entre la partie 220 V et la partie ordinateur du circuit.

PROGRAMME DE DÉMONSTRATION

Si on veut effectuer un défilement de la gauche vers la droite : PA0 vers PA7 soit la lampe 1 vers la lampe 8 nous obtiendrons le tableau suivant :

lampe	8	7	6	5	4	3	2	1	
	$\overline{PA7}$	$\overline{PA6}$	$\overline{PA5}$	$\overline{PA4}$	$\overline{PA3}$	$\overline{PA2}$	$\overline{PA2}$	$\overline{PA0}$	
	0	0	0	0	0	0	0	1	=&h1=(1)10
	0	0	0	0	0	0	1	0	=&h2=(2)10
	0	0	0	0	0	1	0	0	=&h4=(4)10
	0	0	0	0	1	0	0	0	=&h8=(8)10
	0	0	0	1	0	0	0	0	=&h10=(16)10
	0	0	1	0	0	0	0	0	=&h20=(32)10
	0	1	0	0	0	0	0	0	=&h40=(64)10
	1	0	0	0	0	0	0	0	=&h80=(128)10

1 : lampe correspondante allumée

0 : lampe correspondante éteinte

PROGRAMME

```
10 Regcontrole = &h313
20 Port A = &h310
30 OUT Regcontrole, &h80
40 restore 110
50 for X=1 to 8
60 READ ALLUME (X)
70 OUT PORT A, ALLUME (X)
80 GOSUB 120
90 NEXT X
100 GOTO 40
110 DATA 1,2,4,8,16,32,64,128
120 FOR I=1 TO 100 NEXT I:RETURN
```

EXPLICATION DU PROGRAMME

ligne 10 : le registre du PIA reçoit son adresse,

ligne 20 : le registre port A du PIA reçoit son adresse,

ligne 30 : port A en sortie,

ligne 40 : rafraîchissement des données,

ligne 50 : huit données à lire,

ligne 60 : lecture de la donnée,

ligne 70 : envoi de la donnée sur le port A,

ligne 80 : petite attente,

ligne 90 : retour tant que toutes les données ne sont pas envoyées,

ligne 100 : retour au début,

ligne 110 : tableau de données,

ligne 120 : temporisation.

PROGRAMME "CHENIL3"

Ce programme permet de composer des séquences lumineuses commandant huit lampes différentes, s'enchaînant les unes derrières les autres, de les lister et de les modifier.

La vitesse d'enchaînement des séquences est réglable.

Composer RUN "CHENIL3" sous BASIC.

Le programme donne le choix entre :

Entrée séquence=1, modifs=2, listing=3, chenillard=4, retour BASIC=5.

En appuyant sur 1, on va à la composition des séquences lumineuses.

Le programme demande alors : "Nombre de séquences ?"

Par exemple composer 16 puis touche entrée (*enter*), (ce qui signifie qu'il y aura seize séquences lumineuses qui s'enchaîneront les unes derrière les autres avec retour à la première après défilement des seize séquences et ainsi de suite).

Le programme demande la valeur de la première séquence (en décimal de 0 à 255 : 1 allume la première lampe, 2 la deuxième, 4 la troisième, 8 la quatrième, 16 la cinquième, 32 la sixième, 64 la septième et 128 la huitième). Pour allumer plusieurs lampes, il faudra additionner les valeurs propres à chaque lampe (129=1+128 allumera la première et la huitième lampe, 255 allume toutes les lampes 1+2+4+8+16+32+64+128=255.

Composer 1 pour allumer la première lampe puis entrée (enter).

Le programme rappelle la valeur de la première séquence et demande d'entrer la deuxième.

Composer 2 pour la deuxième lampe puis entrée (*enter*),

pour valeur 3, composer 4 pour troisième lampe puis entrée (*enter*),
pour valeur 4, composer 8 pour quatrième lampe puis entrée (*enter*),
pour valeur 5, composer 16 pour cinquième lampe puis entrée (*enter*),
pour valeur 6, composer 32 pour sixième lampe puis entrée (*enter*),
pour valeur 7, composer 64 pour septième lampe puis entrée (*enter*),
pour valeur 8, composer 128 pour huitième lampe puis entrée (*enter*),
pour valeur 9, composer 64 pour septième lampe puis entrée (*enter*),
pour valeur 10, composer 32 pour sixième lampe puis entrée (*enter*),
pour valeur 11, composer 16 pour cinquième lampe puis entrée (*enter*),
pour valeur 12, composer 8 pour quatrième lampe puis entrée (*enter*),
pour valeur 13, composer 4 pour troisième lampe puis entrée (*enter*),
pour valeur 14, composer 2 pour deuxième lampe puis entrée (*enter*),
pour valeur 15, composer 1 pour première lampe puis entrée (*enter*).

Le programme émet alors un bip sonore pour vous indiquer que l'entrée suivante sera la dernière. Composer 0 pour avoir toutes les lampes éteintes.

RETOUR AUTOMATIQUE À LA SÉQUENCE CHOIX

Composer 3 : le listing de vos seize séquences apparaît à l'écran.

Appuyer sur barre espace pour revenir au choix.

Composer 4 : vos séquences défilent les unes derrière les autres avec leur numéro et la valeur correspondante. Les huit lampes s'allument l'une après l'autre en donnant un effet d'aller-retour (*fig.* 13.6).

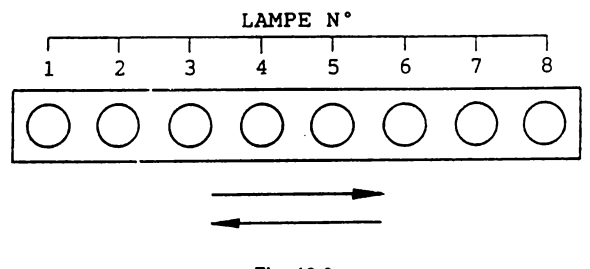


Fig. 13.6

En appuyant sur la touche +, la vitesse de défilement s'accélère, sur la touche –, elle se ralenti.

On peut voir, en ralentissant avec la touche –, que la seizième séquence éteint toutes les lampes. Si on veut par exemple qu'elles s'allument toutes :

— appuyer sur barre espace,

— composer 2 pour modifier.

Le programme vous demande le numéro de la séquence à modifier.

Composer par exemple 14. Le programme affiche le contenu de 13 et de 14 et demande si on veut modifier 14. Appuyer sur touche entrée (enter) pour passer à 15 sans modifier 14, encore touche entrée (enter) pour passer à 16 sans modifier 15.

Composer 255 pour avoir toutes les lampes allumées en séquence 16 puis entrée (enter). Le programme émet un bip pour indiquer que cette séquence est la dernière. Composer – 1 pour retourner au choix.

Composer 3 pour avoir le listing. 16 contient bien 255.

Appuyer sur barre espace retour au choix puis 4.

Les séquences défilent et en 16, on peut voir que toutes les lampes s'allument.

Barre d'espacement pour retour au choix.

Composer 5 pour retour au Basic.

PROGRAMME

```
10 '--programme--cheni13
20 CLS:KEY OFF:COLOR 7,0:OUT &H313,&H82:PORTA=&H310
30 NOMSEQ=15:DIM CHENIL(NOMSEQ)
40 GOTO 110
50 ERASE CHENIL
60 LOCATE 10,27:PRINT" Nombre de séquences :";:INPUT NOMSEQ
70 IF NOMSEQ<1 OR NOMSEQ>9999 THEN BEEP:GOTO 60
80 DIM CHENIL(NOMSEQ)
90 GOTO 170
100 '---entrée--séquences--
110 LOCATE 14,7:COLOR 0,7
120 PRINT " Entrée séquences=1,modifs=2,listing=3,chenillard=4,retour BASIC=5 ";
130 COLOR 7,0
140 DECIDE$=INKEY$:DECIDE=VAL(DECIDE$):IF DECIDE <1 OR DECIDE>5 GOTO 110
150 BEEP
160 ON DECIDE GOTO 50,300,500,660,870
170 CLS
180 FOR I= 1 TO NOMSEQ
190 IF I=NOMSEQ THEN BEEP
200 LOCATE 6,16:PRINT " programme de ";NOMSEQ;" séquence(s) "
210 LOCATE 14,20:PRINT "valeur de";:PRINT USING"####";I;:PRINT"       ";
220 LOCATE,34:INPUT CHENIL(I):IF CHENIL(I)<0 OR CHENIL(I)>255 GOTO 210
230 CLS
240 IF I-1 >=1 THEN LOCATE 10,20 ELSE GOTO 260
250 PRINT "valeur de";:PRINT USING"####";I-1;:PRINT USING"#####";CHENIL(I-1);
260 LOCATE 12,20
270 PRINT "valeur de";:PRINT USING"####";I;:PRINT USING"#####";CHENIL(I);
280 NEXT I
290 CLS:GOTO 110
300 '-----modifs-----
310 CLS
```

```
320 LOCATE 12,20:PRINT" Adresse 1ère séquence:          maximum";NOMSEQ;
330 LOCATE ,44:INPUT ADRESSE
340 IF ADRESSE<0 OR ADRESSE >NOMSEQ GOTO 320
350 I=ADRESSE:CLS
360 LOCATE 25,30:PRINT "Retour menu composer -1";
370 LOCATE 6,16:PRINT " programme de ";NOMSEQ;" séquence(s) "
380 IF I-1>1 THEN LOCATE 10,20 ELSE GOTO 400
390 PRINT "valeur de";:PRINT USING"####";I-1;:PRINT USING"#####";CHENIL(I-1);
400 LOCATE 12,20
410 PRINT "valeur de";:PRINT USING"####";I;:PRINT USING"#####";CHENIL(I);
420 LOCATE 14,20:PRINT "valeur de";:PRINT USING"####";I;:PRINT"     ";
430 LOCATE,34:INPUT CHENIL:IF CHENIL<-1 OR CHENIL(I)>255 GOTO 400
440 IF CHENIL=0 THEN GOTO 470
450 IF CHENIL=-1 THEN CLS:GOTO 110
460 CHENIL(I)=CHENIL
470 IF I=NOMSEQ THEN BEEP:GOTO 490
480 I=I+1
490 CLS:GOTO 360
500 '-----listing----
510 CLS
520 POSI=1:LIGNE=1
530 FOR I=1 TO NOMSEQ
540 LOCATE LIGNE,POSI:PRINT USING"####";I;
550 COLOR 0,7: PRINT USING"####";CHENIL(I);:COLOR 7,0
560 IF LIGNE=24 THEN POSI=POSI+10:LIGNE =0
570 IF POSI>80 THEN LIGNE=0 ELSE GOTO 620
580 LOCATE 25,20:COLOR 31,0:PRINT" Pour continuer appuyer sur barre espace ";
590 COLOR 7,0:POSI=1
600 IF INKEY$=" " GOTO 610 ELSE GOTO 600
610 LOCATE ,10:PRINT SPC(50);:CLS
620 LIGNE=LIGNE+1
630 IF INKEY$=" " THEN INPUT A
640 NEXT I
650 GOSUB 840
660 '-----chenillard--
670 CLS
680 TOT=100
690 LOCATE 25,22,0:COLOR 31,0:PRINT" Pour retour appuyer sur barre espace ";
700 COLOR 7,0
710 FOR I=1 TO NOMSEQ
720 OUT PORTA,CHENIL(I)
730 LOCATE 10,36:PRINT"         ";:LOCATE,30:PRINT USING"####";I;
740 PRINT USING"######";CHENIL(I);
750 FOR ATT=1 TO TOT
760 ENT$=INKEY$
770 IF ENT$=" " THEN CLS:LOCATE 1,1,1:GOTO 110
780 IF ENT$="-" THEN TOT=TOT+5
790 IF TOT<10 GOTO 810
800 IF ENT$="+" THEN TOT =TOT-5
810 NEXT ATT
820 NEXT I
830 GOTO 710
840 LOCATE 25,20:COLOR 31,0:PRINT" Pour continuer appuyer sur barre espace ";
850 COLOR 7,0
860 IF INKEY$=" " THEN CLS:RETURN 110 ELSE GOTO 860
870 COLOR 7,0:CLS:OUT PORTA,0:END
```

⑭ Visualisation de battements cardiaques

PRINCIPE

Chaque battement de cœur entraîne une variation du flux sanguin, ce qui a pour effet de changer l'opacité des vaisseaux sanguins.

La réflexion sur le bout du doigt d'un faisceau infrarouge vers un phototransistor captera donc tous les battements cardiaques.

FONCTIONNEMENT

Le premier ampli opérationnel sert à amplifier le signal et à adapter la tension aux bornes du phototransistor afin d'obtenir un gain optimal du montage.

Le deuxième ampli opérationnel détecte les variations lentes du signal. On obtient donc en sortie un signal carré pulsé par les impulsions cardiaques.

Le troisième ampli opérationnel monté en oscillateur génère un bip sonore matérialisant ainsi le rythme cardiaque.

La LED émettrice augmente l'intensité du rayonnement infrarouge.

CONNEXION

Deux douilles de sorties permettent de visualiser à l'oscilloscope, ou grâce à un ordinateur (muni de la carte 4E/4S + la carte PIA/PC) le battement cardiaque.

CARACTÉRISTIQUES

Alimentation par pile 9 V.

Consommation maximale 70 mA.

Signal de sortie sur douilles ∅ 4 mm avec protection contre les courts-circuits.

Amplitude du signal de sortie : 9 V.

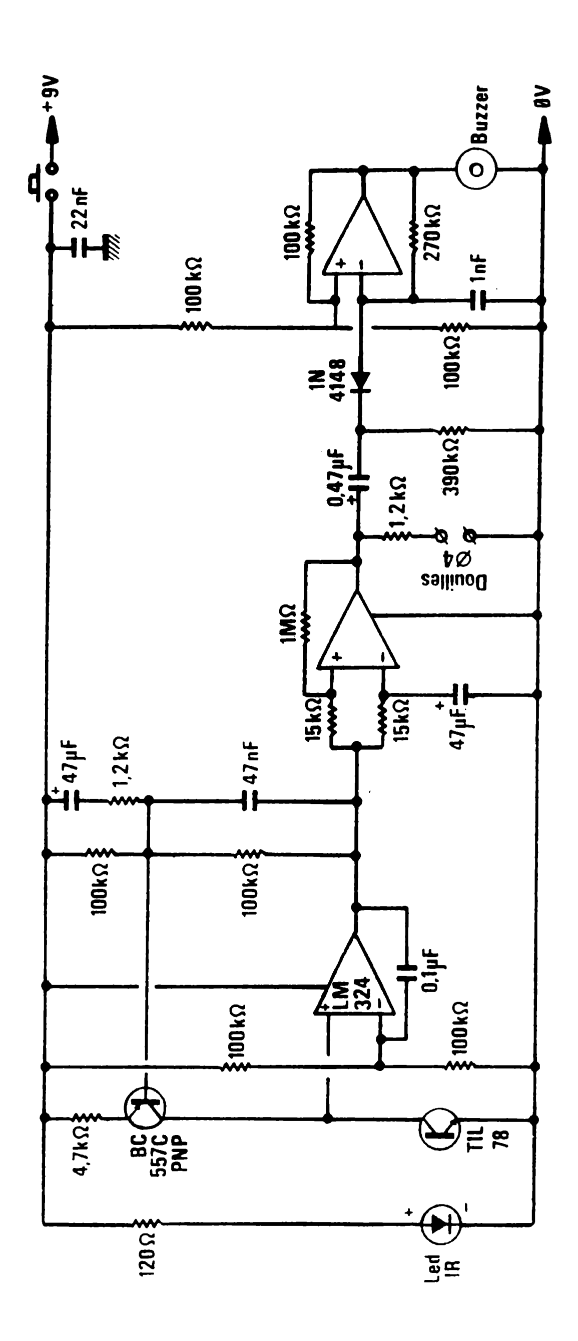

+9V
0V
Buzzer
22nF
100kΩ
100kΩ
270kΩ
1nF
100kΩ
1N
4148
0,47µF
390kΩ
1,2kΩ
Douilles
Ø4
1MΩ
15kΩ
15kΩ
47µF
47µF
1,2kΩ
47nF
100kΩ
100kΩ
LM
324
0,1µF
100kΩ
100kΩ
4,7kΩ
BC
557C
PNP
TIL
78
120Ω
Led
IR

Fig. 14.1

SCHÉMA DE PRINCIPE (*fig.* 14.1)

MONTAGE

Commencer par monter les résistances, les condensateurs, le circuit intégré, la diode 1N 4148, le transistor et le buzzer.

Souder ensuite les deux douilles sur la contre plaque en vérifiant que la face avant soit bien serrée par la douille.

Insérer le phototransistor et la LED infrarouge (en respectant leurs polarités) par le dessus de la face avant de telle manière que le dessus du phototransistor et de la LED soient au ras de la face avant (*fig.* 14.2).

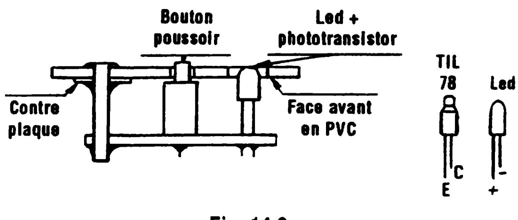


Fig. 14.2

Remarque :

A la mise en marche du montage, attendre quelques secondes avant d'obtenir le bip sonore.

Il s'avère que pour certaine personne le montage ne fonctionne pas correctement. Cela est dû à une mauvaise circulation sanguine dans l'extrémité des doigts.

RACCORDEMENT À LA CARTE 4E/4S (*fig.* 14.3)

Relier grâce à deux cordons munis de douilles ∅ 4 mm à une extrémité et dénudés à l'autre extrémité :

— la douille plus du montage à l'entrée 1 repérée par un plus de la carte 4 entrées/4 sorties,

— l'entrée moins du montage à l'entrée E_1 de la carte 4 entrées/4 sorties.

Maintenir le bouton poussoir appuyé avec un doigt. Un deuxième doigt sera placé sur le trou de diamètre 11 sans l'appuyer trop fort ce qui bloquerait la circulation sanguine.

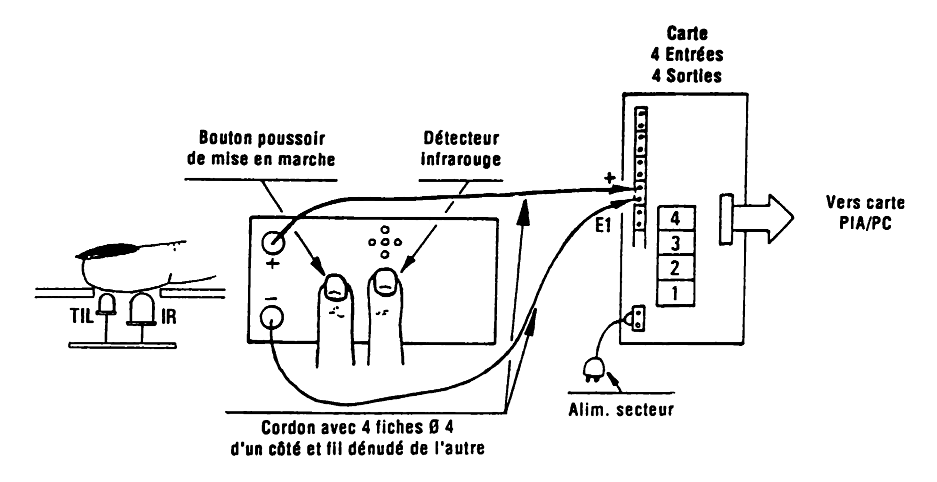


Fig. 14.3

CIRCUIT CÔTÉ CUIVRE (*fig.* 14.4) ET IMPLANTATION (*fig.* 14.5)

Contre plaque

Fig. 14.4

CLIPS PILE
Fil noir
Fil rouge

Fig. 14.5

PERÇAGE DE LA FACE AVANT (*fig.* 14.6)

3 pour la sortie du buzzer,
5 pour les douilles diamètre 4 mm,
6,5 pour le bouton poussoir,
11 pour la Led IR et le phototransistor.

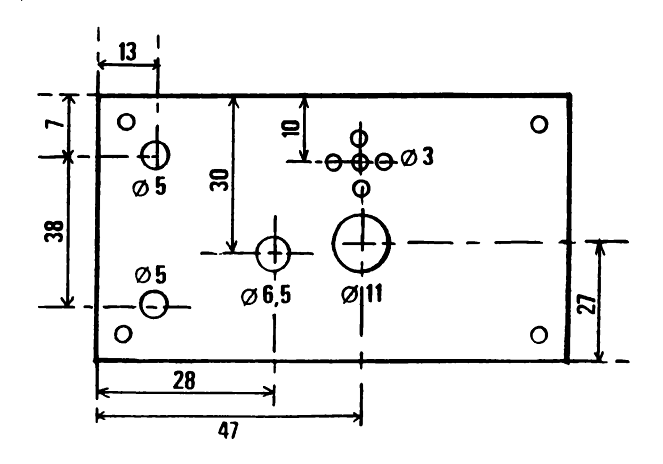


Fig. 14.6

LISTE DES COMPOSANTS

Résistances :

1 × 120 Ω (marron rouge marron),
2 × 1,2 kΩ (marron rouge rouge),
1 × 4,7 kΩ (jaune violet rouge),
2 × 15 kΩ (marron vert orange),
7 × 100 kΩ (marron noir jaune),
1 × 270 kΩ (rouge violet jaune),
1 × 390 kΩ (orange blanc jaune),
1 × 1 MΩ (marron noir vert).

Condensateurs :

2 × 47 µF/25 V radial,
1 × 0,47 µF/50 V radial,
1 × 0,1 µF,
1 × 47 nF céramique,
1 × 22 nF céramique,
1 × 1 nF céramique.

Semiconducteurs :

1 × LM 324,
1 × TIL 78 (phototransistor),
1 × Led IR émettrice,
1 × BC 557 C (impératif) (T),
1 × 1N 4148 (D).

Divers :

un buzzer,
deux douilles diamètre 4 mm,
un bouton poussoir rouge,
un clips de pile,
un boîtier noir ref. 10P.

PROGRAMME CŒUR1

Ce programme simple écrit en Basic permet de visualiser par un bip micro, ainsi qu'un affichage écran le rythme cardiaque.

Pour lancer ce programme, faire RUN "CŒUR1" Pour arrêter le programme, actionner la touche barre espace.

```
10  CLS:KEY OFF
20  OUT &H313,&H82
30  PORT.B=&H311
40  LOCATE 2,19:PRINT "************ Cardiotachymétre ************"
50  LOCATE 6,24:PRINT "Calculez votre rythme cardiaque"
60  LOCATE 8,19:PRINT "Nonbre de pulsations / minute optimal = 80"
70  LOCATE 20,25:PRINT "(Barr. espace) pour sortir"
80  IF INP(PORT.B) = 1 GOTO 100
90  A$=INKEY$:IF A$=" " THEN END ELSE 80
100 LOCATE 14,33:PRINT "          "
110 IF INP(PORT.B) = 0 GOTO 130
120 A$=INKEY$:IF A$=" " THEN END ELSE 110
130 BEEP:LOCATE 14,33:PRINT "Boum Boum!":GOTO 80
```

EXPLICATION DU PROGRAMME

lignes 10 à 30 : ces instructions permettent d'effacer l'écran, d'initialiser la carte PIA/PC Port A en sortie, Port B en entrée et de définir l'adresse du Port B (voir notice carte PIA/PC),

lignes 40 à 90 : tant que le signal de l'entrée 1 reste au niveau bas, le programme va à la ligne suivante (ligne 90), et si la touche barre espace n'est pas actionnée, le programme retourne à la

ligne 80. L'action de la touche barre espace entraîne l'arrêt du programme et la détection d'un front montant (niveau haut) sur l'entrée 1 entraîne un branchement à la ligne 100,

ligne 100 : efface l'inscription de la ligne 130,

lignes 110 à 120 : le fonctionnement est identique aux instructions lignes 80 et 90 sauf pour la détection du signal d'entrée qui s'effectue sur un front descendant et pour le rebouclage du programme à ligne 110 qui s'effectue sur un niveau haut,

ligne 130 : création d'un bip sonore et affichage à l'écran d'un message, puis retour au début du programme ligne 80.

⑮ Bloc deux afficheurs

TECHNOLOGIE

Ce montage très simple permet, grâce à deux afficheurs sept segments et un convertisseur BCD/7 segments, la mise en évidence du multiplexage et le principe de l'affichage.

SCHÉMA DE CÂBLAGE (*fig.* 15.1)

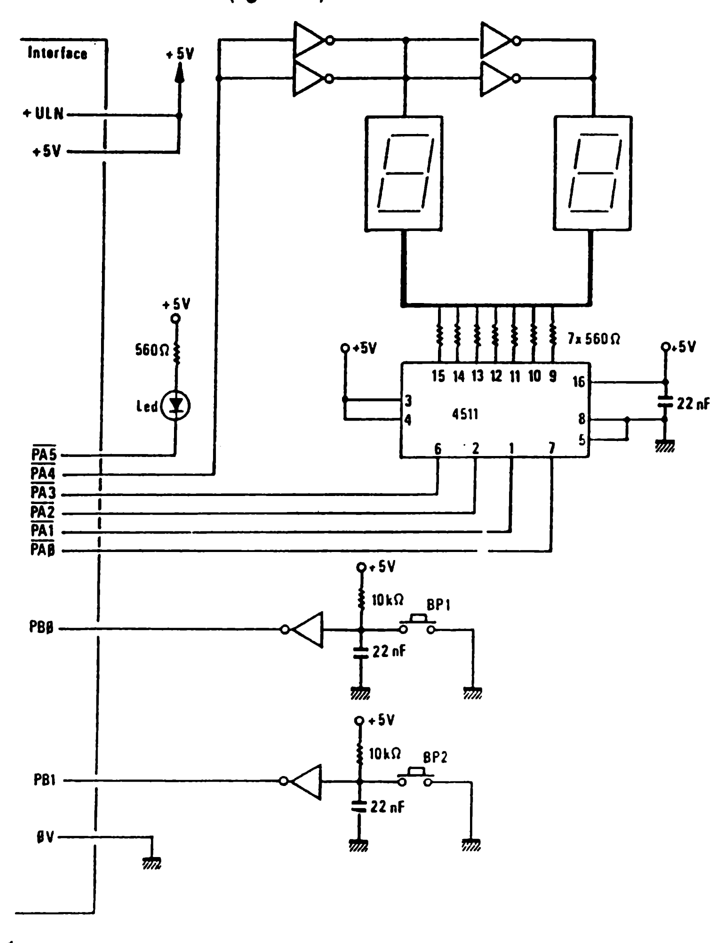


Fig. 15.1

CONNEXION

Ce montage peut être connecté sur tout ordinateur PC, par l'intermédiaire de la carte PIA/PC. Du fait de sa faible consommation, son alimentation est fournie par le PC.

SCHÉMA DE PRINCIPE (*fig.* 15.2)

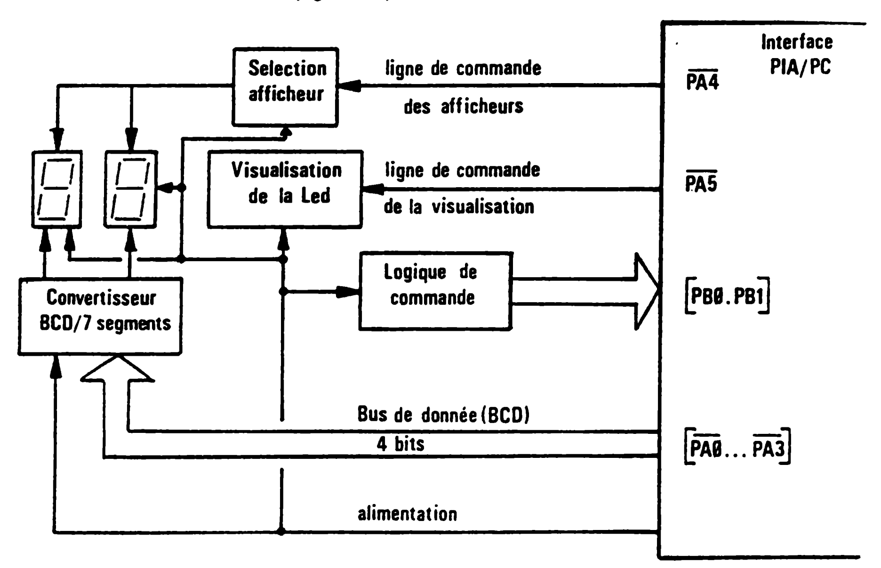


Fig. 15.2

LES AFFICHEURS

Ce sont des afficheurs 7 segments à cathodes communes.

Les segments sont référencés comme indiqué sur la figure 15.3a.

Pour afficher le chiffre 2, il faudra donc commander les segments : a, b, d, e, g (*fig.* 15.3b).

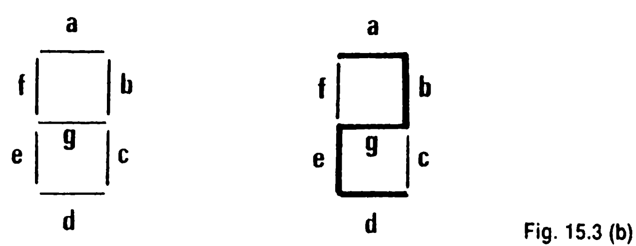


Fig. 15.3 (a)

Fig. 15.3 (b)

Tableau des segments suivant les valeurs BCD à afficher

affichage	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	0	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	0	0	1	1

0: segment éteind
1: segment allumé

----- chiffre 2

CONVERSION BCD/7 SEGMENTS

Le convertisseur (4511) transforme les valeurs BCD du bus de données en données 7 segments visualisables sur les afficheurs.

Les données BCD reçues par le convertisseur sont complémentées du fait de la constitution de la carte PIA/PC.

Donc, pour envoyer le chiffre 2 aux afficheurs par l'intermédiaire du convertisseur, la donnée BCD sera égale à : 15 – 2 = 13.

VISUALISATION DE LA LED

La commande de la visualisation de la LED s'effectue par la ligne PA5 du port A du PIA :

— si PA5 = 0 : la LED est éteinte

— si PA5 = 1 : la LED est allumée.

LOGIQUE DE COMMANDE

Cette fonction est réalisée par 2 boutons poussoirs : BP1 et BP2. Ceux-ci permettent la prise en compte par le logiciel d'événements extérieurs, car ils sont reliés respectivement à PB0 et PB1 du Port B du PIA :

— si BP1 non appuyé : PB0 = 0

— si BP1 appuyé : PB0 = 1

— si BP2 non appuyé : PB1 = 0

— si BP2 appuyé : PB1 = 1.

SÉLECTION DES AFFICHEURS

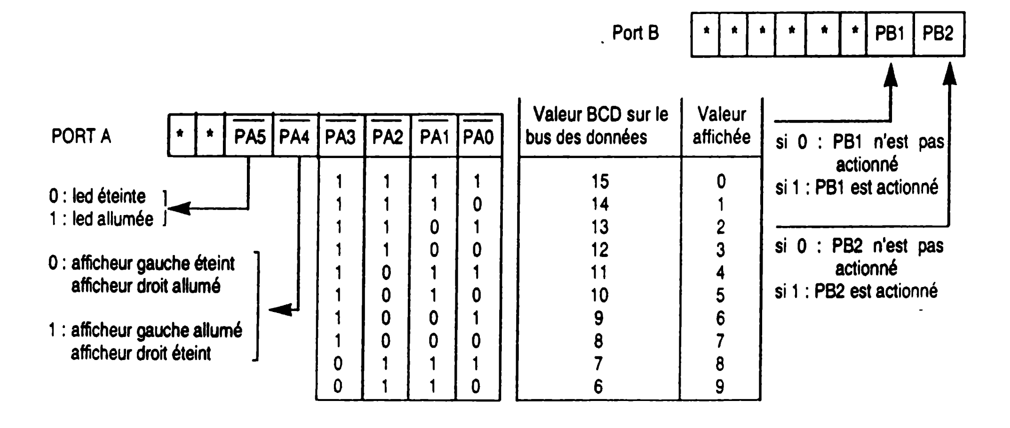
C'est la partie du montage qui est multiplexée : les 2 afficheurs sont commandés par la même ligne : PA4 :

— si PA4 = 0 : l'afficheur de gauche ne sera pas sollicité

l'afficheur de droite s'allume avec la valeur à afficher présente sur le bus de données

— si PA4 = 1 : l'afficheur de gauche s'allume l'afficheur de droite n'est pas sollicité.

RÉCAPITULATIF



$\overline{PA3}$	$\overline{PA2}$	$\overline{PA1}$	$\overline{PA0}$	Valeur BCD sur le bus des données	Valeur affichée
1	1	1	1	15	0
1	1	1	0	14	1
1	1	0	1	13	2
1	1	0	0	12	3
1	0	1	1	11	4
1	0	1	0	10	5
1	0	0	1	9	6
1	0	0	0	8	7
0	1	1	1	7	8
0	1	1	0	6	9

CIRCUIT IMPRIMÉ (*fig.* 15.4) ET IMPLANTATION (*fig.* 15.5)

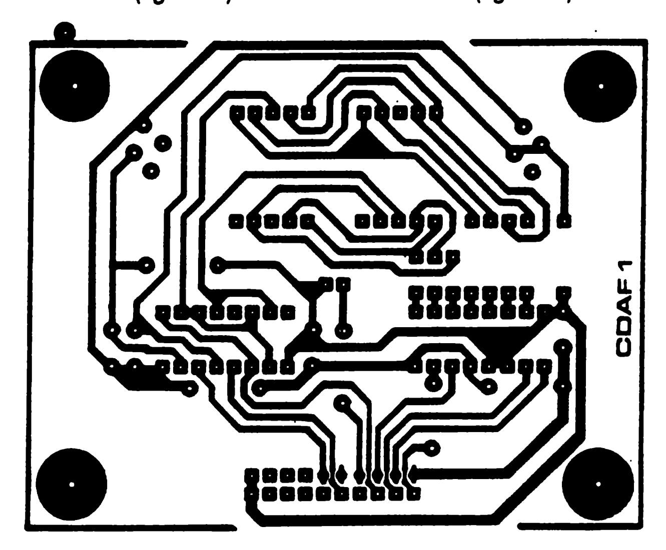


Fig. 15.4

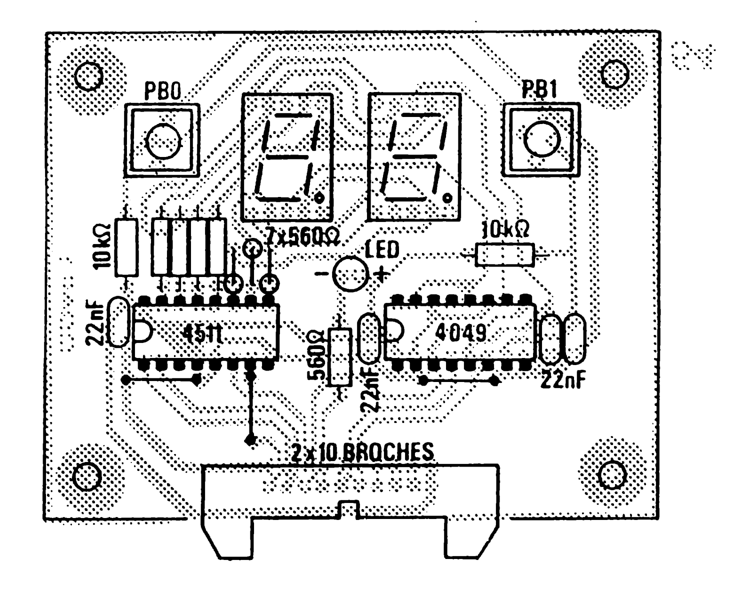


Fig. 15.5

MONTAGE

Souder dans l'ordre et le plus près possible du circuit :

— les trois straps repérés,

— les dix résistances,

— les quatre condensateurs,

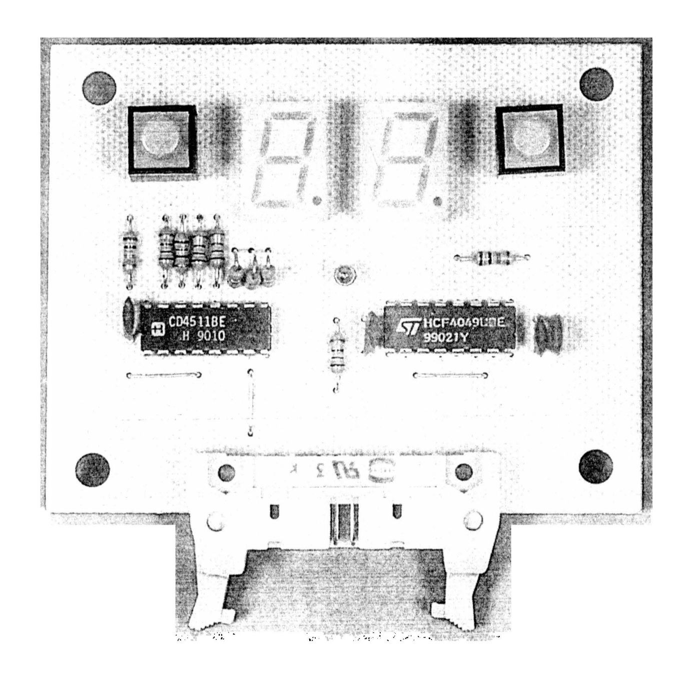
— la LED : attention à la polarité, patte longue au plus,

— les deux afficheurs avec le point décimal en bas à droite,

— les deux touches pour CI,

— les circuits intégrés repérés par un point ou une encoche,

— le connecteur informatique.



LISTE DES COMPOSANTS

8 × 560 Ω (vert bleu marron),
2 × 10 kΩ (marron noir orange),
4 × 22 nF,
1 × led rouge diamètre 3,
1 × 4511,
1 × 4049,
2 × afficheur cathodes communes,
2 × touche pour CI.
1 × connecteur mâle coudé 2 × 10 br.

PROGRAMME AFF1

PROGRAMME AFF1 SOUS BASIC

Allumer le nombre 3 sur le premier afficheur.

```
10 '__programme__aff1
20 CLS
30 PORT.A     = &H310
40 PORT.B     = &H311
50 REG.CON    = &H313
60 MODE.CONF = &H82
70 OUT REG.CON,MODE.CONF
80 LOCATE 10,10:PRINT"Appuyer sur barre espace pour arrêt "
90 NOMBRE     = 3
100 VAR.U=15-NOMBRE
110 OUT PORT.A,VAR.U
120 SP$=INKEY$ : IF SP$<>" " THEN GOTO 110
130 OUT PORT.A,0
140 END
```

EXPLICATION DU PROGRAMME

lignes 30 à 60 : configuration du PIA,

ligne 90 : nombre = 3,

ligne 100 : complément à 15 de nombre dans VAR.U,

ligne 110 : sortie sur Port A de VAR.U,

ligne 120 : si entrée espace - - - fin, sinon retour à 110,

ligne 130 : éteindre afficheur,

ligne 140 : fin.

PROGRAMME AFF2

PROGRAMME AFF2 SOUS BASIC

Allumer les deux afficheurs avec le nombre 13.

```
10 '___programme__aff2
20  CLS
30 '__config__pia
40  PORT.A    = &H310
50  PORT.B    = &H311
60  REG.CON   = &H313
70  MODE.CONF = &H82
80  OUT REG.CON,MODE.CONF
90 '__NOMBRE__AFFICH__=__13
100 LOCATE 10,10:PRINT " Appuyer sur barre espace pour arrêt "
110 CHIFFRE1  = 3
120 CHIFFRE2  = 1
130 VAR(1)=15-CHIFFRE1
140 VAR(2)=15-CHIFFRE2
150 OUT PORT.A,B+VAR(I)
160 '_B=0__AFFICH__3__=16__AFFICH__1
170 IF B=16 THEN B=0 :I=1:GOTO 190
180 IF B=0 THEN B=16:I=2
190 SP$=INKEY$ : IF SP$<>" " THEN GOTO 150
200 OUT PORT.A,0
210 END
```

EXPLICATION DU PROGRAMME

lignes 30 à 80 : configuration PIA,

lignes 110 et 120 : transfert dans chiffre 1 = 3, chiffre 2 = 1,

lignes 130 et 140 : complément chiffre 1 et chiffre 2 à 15 dans VAR(1) et VAR(2),

ligne 150 : sortie sur port A (constante allumage chiffre dans B, valeur du chiffre dans VAR(I).

ligne 170 : test si deuxième chiffre, changement constante d'allumage du premier chiffre dans B, constante d'adressage premier chiffre dans I. Aller à 190,

ligne 180 : test si premier chiffre, chargement constante d'allumage deuxième chiffre dans B, constante d'adressage deuxième chiffre dans I. Aller à 190.

ligne 190 : si barre espace non appuyée, aller à 150,

ligne 200 : éteindre afficheur,

ligne 210 : fin.

PROGRAMME AFF3

PROGRAMME AFF3 SOUS BASIC

Si on appuie sur la touche de gauche, on compte les secondes, sinon décomptage par la touche de droite.

```
10 '__programme__aff3
20  CLS
30  PORT.A=&H310
40  PORT.B=&H311
50  REG.CON=&H313
60  MODE.CONF = &H82
70  OUT REG.CON,MODE.CONF
80  PRINT"Comptage : BP de Gauche ; Décomptage : BP de Droite."
90  PRINT"       <BARRE ESPACE> pour sortir du programme."
100 M=0:I=1
110 IF (INP(PORT.B) AND 3)=0 THEN 110
120 IF (INP(PORT.B) AND 1)=1 THEN F=0:GOTO 120
130 IF (INP(PORT.B) AND 2)=2 THEN F=1:GOTO 130
140 IF F=0 THEN 190 ELSE 150
150 VAR.U=VAR.U-1
160 IF VAR.U < 0 THEN VAR.U=9 : VAR.D=VAR.D-1
170 IF VAR.D < 0 THEN VAR.U=0 : VAR.D=0 : OUT PORT.A,32 : GOTO 120
180 GOTO 220
190 VAR.U=VAR.U+1
200 IF VAR.U > 9 THEN VAR.U=0 : VAR.D=VAR.D+1
210 IF VAR.D > 9 THEN VAR.U=9 : VAR.D=9 : OUT PORT.A,32 : GOTO 120
220 A$=RIGHT$(TIME$,2)
230 VAR(1)=15-VAR.U : VAR(2)=15-VAR.D
240 SP$=INKEY$ : IF SP$=" " THEN OUT PORT.A,0 : CLS : END
250 OUT PORT.A,VAR(I)+M
260 IF M=16 THEN M=0:I=1:GOTO 280
270 IF M=0 THEN M=16:I=2:GOTO 280
280 B$=RIGHT$(TIME$,2):IF A$=B$ GOTO 240 ELSE GOTO  120
```

EXPLICATION DU PROGRAMME

lignes 30 à 70 : configuration PIA,

ligne 100 : remise à 0 constante d'adressage afficheur 1 ou 2 (M), I = 1 constante premier chiffre,

lignes 110 à 130 : test des touches du bloc afficheur, si touche 1 F = 0, touche 2, F = 1,

lignes 140 à 160 : décomptage,

ligne 170 : si fin de décomptage, allumage de la Led (OUT Port A, 32),

ligne 180 : aller à 200,

lignes 190 à 210 : comptage avec allumage de la LED après 99,

ligne 220 : A$ = valeur des secondes de l'horloge du micro,

ligne 230 : complémentation à 15 de U et D, résultat dans VAR(1) et VAR(2),

ligne 240 : si espace composé, éteindre afficheur, fin,

ligne 250 : sortie sur Port A valeur de VAR(I) I = 1 ou 2 + M (M adressage afficheur 1 ou 2 = 0 ou 16,

lignes 260 et 270 : changement d'afficheur et de chiffre,

ligne 280 : chargement dans B$ de la valeur des secondes de l'horloge du micro. Si A$ = B$, les secondes n'ont pas changées.

Retour à 240 pour changer d'afficheur, sinon, retour à 120.

⑯ Digivolt
un voltmètre connecté sur un PC

Le but de cette réalisation est d'obtenir un voltmètre 3 digits (1 000 points) permettant de transformer une mesure de tension continue positive en un nombre de trois chiffres en BCD (Binaire Codé Décimal) directement exploitable par l'ordinateur. De même, la mesure obtenue sera en permanence affichée sur le voltmètre, de façon à pouvoir se servir de celui-ci en autonome.

Le voltmètre ainsi réalisé se connectera directement sur la carte PIA/PC.

DESCRIPTION DE L'APPAREIL

Il comporte trois gammes de mesures :

0 à 999 mV résolution 1 mV,
0 à 9,99 V résolution 10 mV,
0 à 99,9 V résolution 100 mV.

Ces gammes sont sélectionnées par un sélecteur de gammes (commutateur rotatif).

Un dépassement de gamme produit l'affichage de "EEE" :

— l'impédance d'entrée est de 1 MΩ,

— entrée protégée jusqu'à une tension de 220 V,

— vitesse de mesure maximale 90 m par seconde. Cette vitesse dépend en partie du type de PC sur lequel sera connecté le voltmètre, du fait de la fréquence d'horloge du micro ainsi que du langage utilisé,

— alimentation 220 V 5 VA, protection par fusible.

SYNOPTIQUE (*fig.* 16.1)

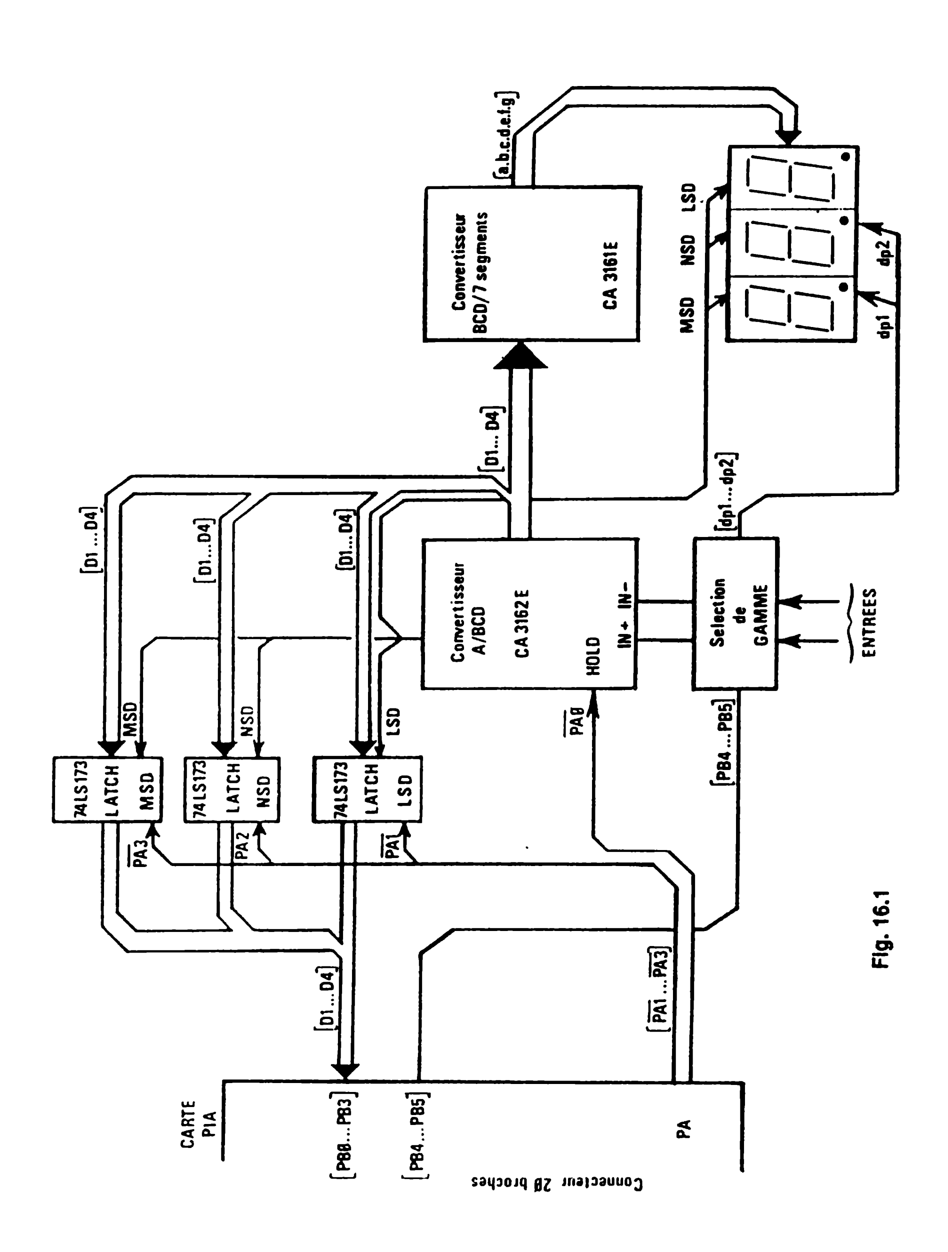


Fig. 16.1

CONVERTISSEUR ANALOGIQUE/BINAIRE CODÉ DÉCIMAL

Le CA 3162 permet la conversion d'une tension continue en Binaire Codé Décimal. La lecture en BCD de cette tension s'effectue sur le bus de données (D1 à D4) D1 bit de poids faible à D4 bit de poids fort.

L'entrée Hold sur le 3162 mise à niveau haut bloque la prise de mesure à l'entrée de celui-ci, de façon à pouvoir effectuer la lecture (la mesure étant alors stable).

DÉCODEUR BCD/7 SEGMENTS

Le décodeur (CA 3161) reçoit le code BCD (D1 à D4) en provenance du CA 3162 et le convertit en un code 7 segments appliqué ensuite directement sur les afficheurs.

Trois signaux MSD, NSD, LSD assurent le multiplexage des afficheurs.

L'affichage est réalisé par 3 afficheurs 7 segments à Leds. Les points "." de chaque afficheur sont gérés électriquement par la sélection de gammes.

SÉLECTION DE GAMMES

À l'aide d'un pont diviseur de tension et d'un commutateur, on sélectionne les trois gammes tout en positionnant le "." sur les afficheurs et en informant l'ordinateur de la gamme choisie (entrées 4 et 5 du port B).

LES LATCHS

Les trois circuits 74 LS 173 sont chargés par les données en provenance du CA 3162 (D1...D4). La sélection du LATCH étant assurée, comme pour les afficheurs, par MSD, NSD, LSD.

La lecture des LATCHS est effectuée ensuite grâce aux poids (PA1, PA2, PA3) du port A (étant haut actif). PA0 assure le blocage de la mesure (Hold) sur le 3162. Les données BCD contenues dans les LATCHS sont transférées sur le port B (PB0 à PB3) au passage à l'état 1 (haut) de PA1 à PA3 sélectionnant le 1er, le 2e et le 3e chiffre du nombre mesuré.

SÉLECTION DE LA LECTURE DE LA MESURE

Cette lecture de la valeur mesurée par le voltmètre s'effectue grâce au port A de l'interface configuré en sortie, le port B étant lui configuré en entrée.

Port A en sortie

	$\overline{PA7}$	$\overline{PA6}$	$\overline{PA5}$	$\overline{PA4}$	$\overline{PA3}$	$\overline{PA2}$	$\overline{PA1}$	$\overline{PA0}$
Port A	NC	NC	NC	NC	MSD	NSD	LSD	Hold

NC : non connecté

LSD : poids faible de la mesure

NSD : poids moyen

MSD : poids fort

Port B en entrée

	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
Port B	NC	NC	DP2	DP1	D4	D3	D2	D1

PB7 et PB6 non connectés.

PB5 et PB4 reçoivent la codification de la gamme sélectionnée à travers le commutateur

PB5	PB4	
0	1	gamme 0 à 9,99 V
1	0	gamme 0 à 99,9 V
1	1	gamme 0 à 999 mV

PB 0 et PB3 reçoivent les données D1 à D4 en BCD.

Séquence de lecture

	$\overline{PA3}$	$\overline{PA2}$	$\overline{PA1}$	$\overline{PA0}$	
1. Port A	0	0	0	0	commande programme $0

$\overline{PA0}$ (Hold) = 0 le voltmètre prend des mesures

2. Port A	0	0	0	1	commande programme $1

$\overline{PA0}$ = 1 blocage de la mesure dans le voltmètre

3. Port A	0	0	1	1	commande programme $3

$\overline{PA0}$ = 1 mesure toujours bloquée

$\overline{PA1}$ = lecture LSD

4. Port A	0	1	0	1	commande programme $5

$\overline{PA0}$ = 1 mesure toujours bloquée

$\overline{PA2}$ = lecture NSD

5. Port A	1	0	0	1	commande programme $9

$\overline{PA0}$ = 1 mesure toujours bloquée

$\overline{PA3}$ = lecture MSD

— La conversion des données par DIGIVOLT se fait en deux temps :

— un temps de conversion analogique digitale d'environ 4 ms (voir figure 16.2),

— un temps de transfert des données issues de cette conversion dans les LATCHS d'environ 16 ms. MSD est d'abord chargé, puis LSD et enfin NSD,

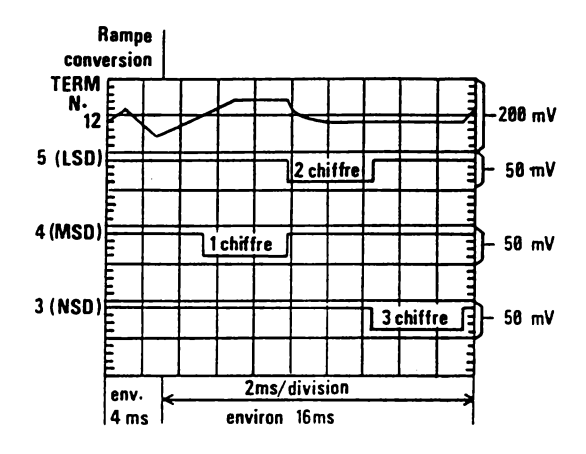
— le programme ne peut aller prélever ces informations que lorsque ce transfert est terminé.

Il faudra donc un temps minimum de 4 ms pendant lequel la prise de mesure est libérée (HOLD du niveau bas) permettant ainsi la conversion de la tension d'entrée. Puis un temps minimum de 16 ms pendant lequel

la mesure est bloquée à l'entrée pour avoir le transfert dans les trois LATCHS de sortie.

Ensuite, seulement l'ordinateur prélève les trois chiffres de la mesure suivante.

Le non-respect de ces temps donne une mesure plusieurs fois à la même valeur (alors que la mesure varie) si le temps de 4 ms n'est pas respecté. Une mesure erronée sur les centaines, les dizaines ou l'unité si le temps de blocage de la mesure d'entrée est trop court (mesure change pendant le transfert à l'ordinateur).



Mode de prise de donnée par le 3162

Fig. 16.2

SCHÉMA (*fig.* 16.3)

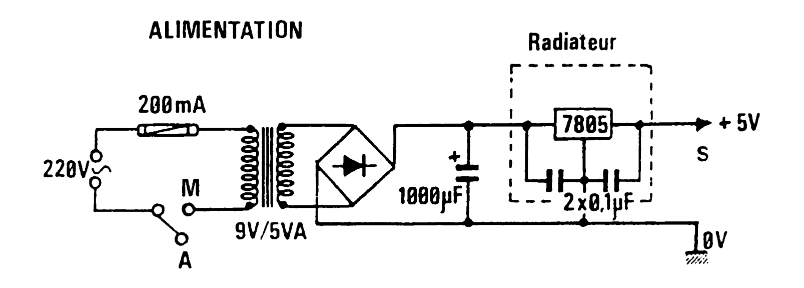


Fig. 16.3a

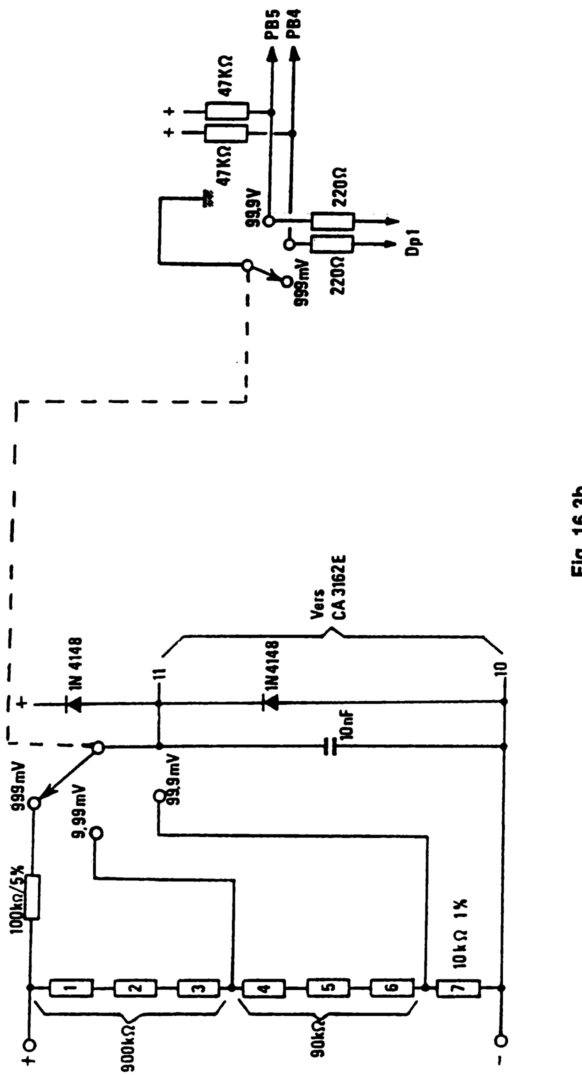

+
900kΩ
90kΩ
10kΩ 1%
-
1
2
3
4
5
6
7
100kΩ/5%
999mV
9,99mV
99,9mV
10nF
1N 4148
1N4148
11
10
Vers
CA 3162E
47KΩ
47KΩ
PB5
PB4
99,9V
999mV
220Ω
220Ω
Dp1

Fig. 16.3b

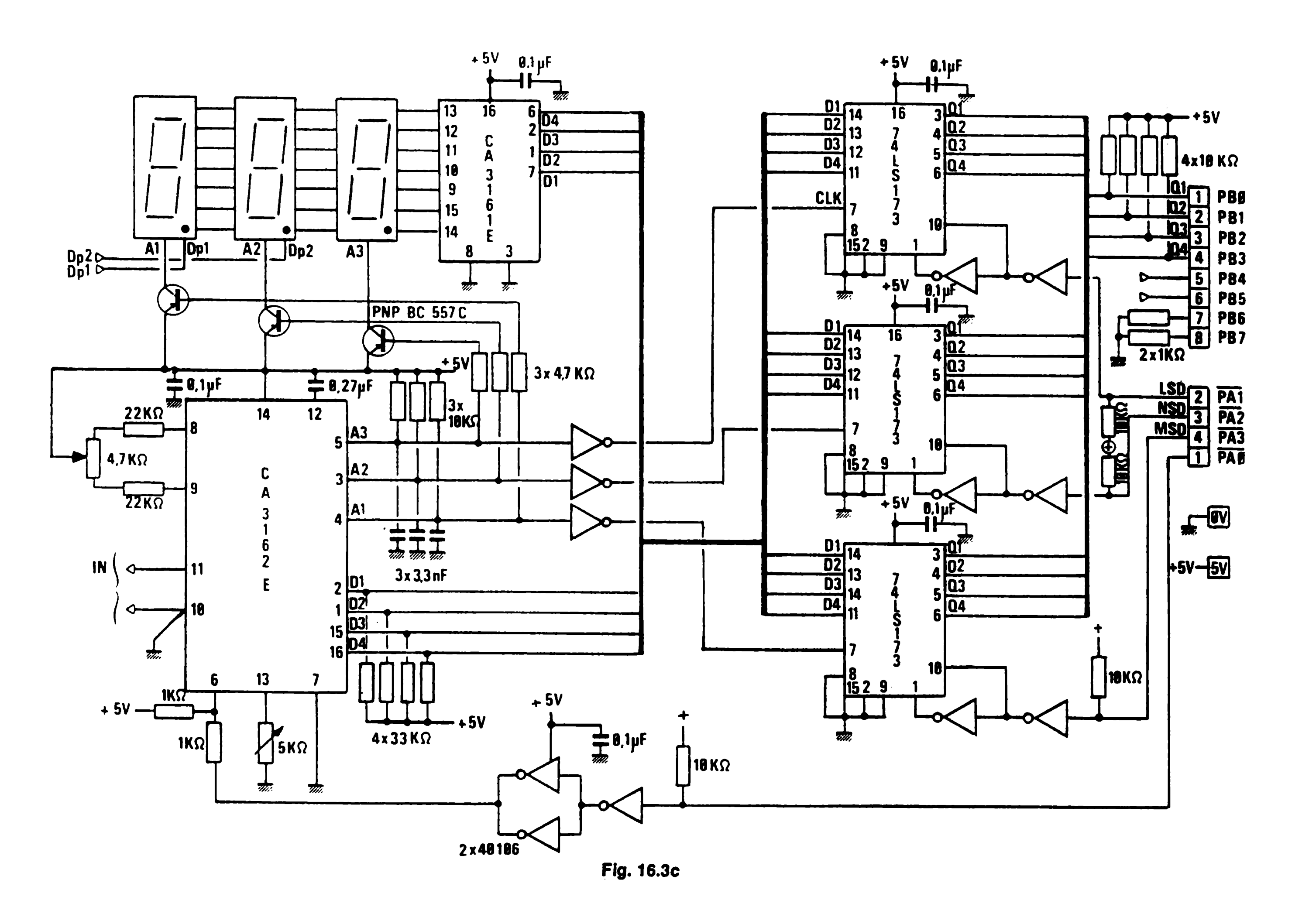

CA 3161 E
CA 3162 E
74LS173
74LS173
74LS173
PNP BC 557 C
3 x 4,7 KΩ
3x 10KΩ
3x 3,3nF
4x 33 KΩ
4x 10 KΩ
2x 1KΩ
2x 40106
22KΩ
4,7KΩ
5KΩ
1KΩ
10KΩ
0,1μF
0,27μF
+5V
IN
CLK
LSD
NSD
MSD
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PA1
PA2
PA3
PA0
0V
5V
Dp1
Dp2
A1
A2
A3
D1
D2
D3
D4
Q1
Q2
Q3
Q4

Fig. 16.3c

CIRCUITS IMPRIMÉS ET IMPLANTATION (*fig.* 16.4 et 16.5)

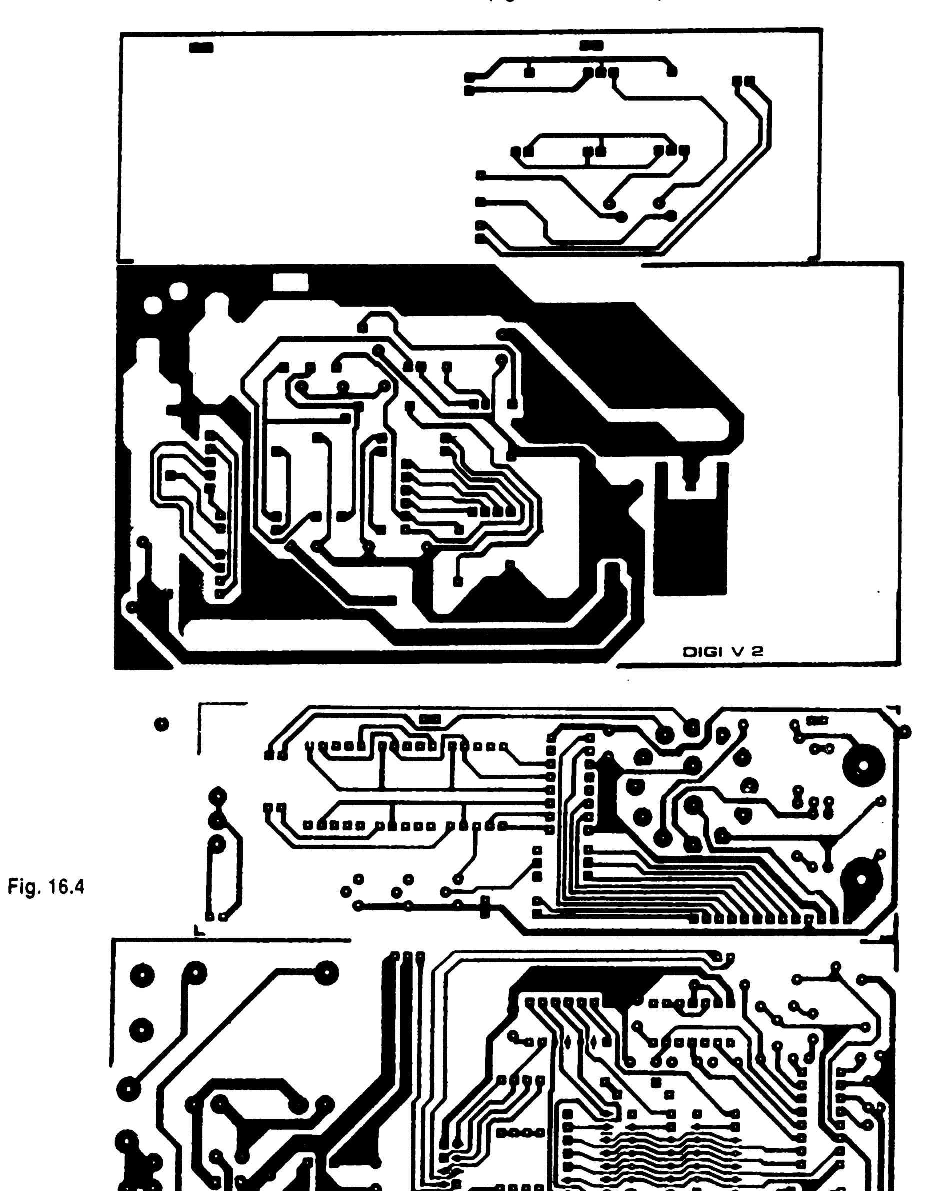


Fig. 16.4

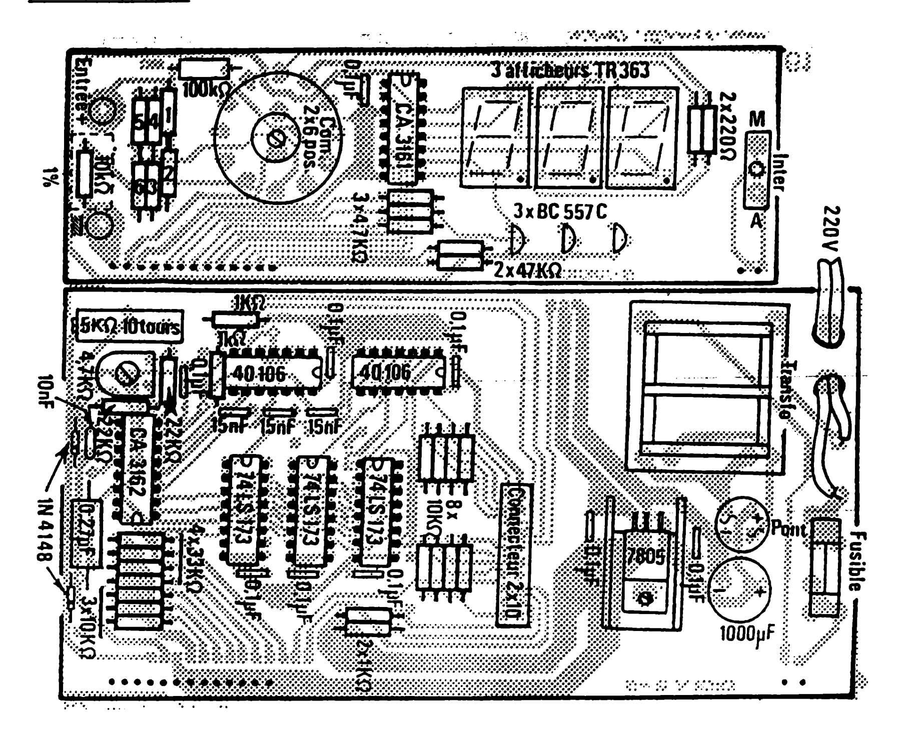


Fig. 16.5

LISTE DES COMPOSANTS

Résistances : 5 %

1 × 100 Ω (marron noir marron),
2 × 220 Ω (rouge rouge marron),
4 × 1 kΩ (marron noir rouge),
3 × 4,7 kΩ (jaune violet rouge),
11 × 10 kΩ (marron noir orange),
2 × 22 kΩ (rouge rouge orange),
4 × 33 kΩ (orange orange orange),
2 × 47 Ω (jaune violet orange),
1 × 100 kΩ (marron noir jaune).

Résistances : 1 %

866 kΩ (gris bleu bleu orange) + 34 kΩ (orange jaune noir rouge),
ou
787 kΩ (violet gris violet orange) + 113 kΩ (marron marron orange orange),
86,6 kΩ (gris bleu bleu rouge) + 3,4 kΩ (orange jaune noir marron),
ou
9,31 kΩ (blanc orange marron marron) + 80,6 kΩ (gris noir bleu rouge),
10 kΩ (marron noir noir rouge).

Semi-conducteurs

1 × pont 1,5 A,
3 × BC 557 C,
1 × régulateur 7805,
3 × 74 LS 173,
2 × 40 106,
1 × CA 3162,
1 × CA 3161,
2 × 1N 4148.

Condensateurs

1 × 10 nF,
3 × 15 nF,
9 × 0,1 µF,
1 × 0,27 µF,
1 × 1 000 µF/25 V radial.

Divers

1 × porte fusible pour CI,
1 × fusible 200 mA,
3 × afficheur anode commune TR 363,
1 × radiateur en U,
1 × vis 3 × 10 + écrou,
1 × commutateur 2 × 6 positions,
1 × inverseur à bascule pour CI,
1 × ajustable 4,7 kΩ horizontale,
1 × ajustable 5 kΩ 10 tours,
1 × transformateur 9 V-5 VA,
2 × rivet 4 × 20,
1 × câble informatique 20 conducteurs,
1 × connecteur femelle 2 × 10 broches,
1 × connecteur 2 × 10 broches pour CI,
1 × bouton à vis pour commutateur,
1 × barrette de connexion coudée 15 broches.

Attention : Sur le schéma d'implantation du circuit afficheur, on remarque que six résistances sont repérées non pas par leur valeur, mais par des numéros de 1 à 6.

La somme des valeurs des résistances 1 + 2 + 3 doit être égale à 900 kΩ.

La somme des valeurs des résistances 4 + 5 + 6 doit être égale à 90 kΩ.

Suivant les valeurs des résistances fournies dans la pochette de composants, vous monterez donc :

Résistances n.

1	2	3
866 kΩ	+ 34 kΩ	+ strap
787 kΩ	+ 133 kΩ	+ strap

ou

4	5	6
86,6 kΩ	+ 3,4 kΩ	+ strap
80,6 kΩ	+ 9,31 kΩ	+ 100 Ω

Suivant le cas, certaines résistances sont remplacées par des straps qui devront être impérativement montés pour que l'appareil fonctionne.

MONTAGE

Attention : les circuits étant réalisés en double face, ne pas oublier les soudures sur le côté composants.

Circuit principal

Souder dans l'ordre suivant et le plus près possible du circuit :

— les résistances

— les condensateurs sauf le 1 000 µF

— le pont de diodes

— les deux ajustables

— le porte fusible

— le condensateur chimique de 1 000 µF (attention à la polarité)

— les circuits intégrés en respectant le sens d'insertion repéré par un point ou une encoche (la soudure doit être rapide pour éviter de trop les chauffer)

— le régulateur 7805 et son radiateur (*fig.* 16.6)

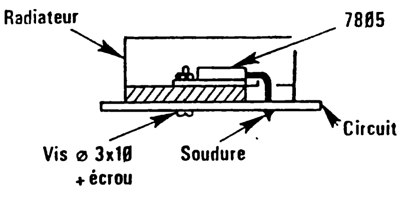


Fig. 16.6

— le transformateur

— le fil en le passant au préalable dans les deux trous prévus pour le blocage de celui-ci

— mettre le fusible en place dans le porte fusible

Circuit affichage

Souder dans l'ordre suivant et le plus près possible du circuit :

— le strap

— les résistances

— les trois afficheurs en plaçant le point décimal en bas à droite (voir implantation)

— les 3 BC 557 C dont le sens d'insertion est repéré par un méplat

— le CA 3161 repéré par un point ou une encoche

— l'interrupteur

— le commutateur 2 × 6 positions en plaçant sa butée sur la 3e position comme expliqué ci-dessous :

dévisser l'écrou du commutateur, retirer la rondelle éventail et la butée. Tourner le commutateur à fond vers la gauche. Remettre la butée en place en positionnant l'ergot sur le 3 (indiqué sur le corps du commutateur). Replacer la rondelle, puis resserrer l'écrou (*fig.* 16.7) :

— les deux douilles en laiton 4 mm,

— coupez la barrette de connexion 15 broches en deux morceaux de 13 et 2 chacun. Soudez-les sur le circuit comme indiqué sur la figure 16.8. Les trous sont repérés sur l'implantation par des rangées de points.



Fig. 16.7 Fig. 16.8

ASSEMBLAGE DES DEUX CIRCUITS

Insérez les picots des barrettes de connexion sur le circuit principal, soudez-les, puis coupez au ras de la soudure (*fig.* 16.9).

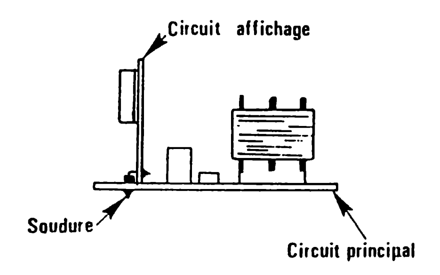


Fig. 16.9

RÉGLAGE

Placer le commutateur sur la gamme 9,99 V.

Relier entre elles avec un cordon les douilles "entrée +" et "⏚". Régler l'ajustable de 4,7 kΩ pour afficher 000 sur le voltmètre. Injecter une tension de 9,9 V environ (mesurée avec un voltmètre étalon) sur les douilles "entrée +" et "⏚" et à l'aide de l'ajustable de 5 kΩ/10 trous, afficher sur votre voltmètre la même valeur que celle lue sur le voltmètre étalon.

Brancher le cordon secteur au 220 V.

Brancher le cordon informatique venant du digivolt sur la carte PIA. Mettre le digivolt sous tension en mettant son interrupteur en position marche.

La tension à mesurer sera à connecter entre les points entrée + et entrée 0 (+ étant le + de la tension à mesurer).

Le digivolt est protégé intérieurement par un fusible 200 mA(*fig.* 16.10).

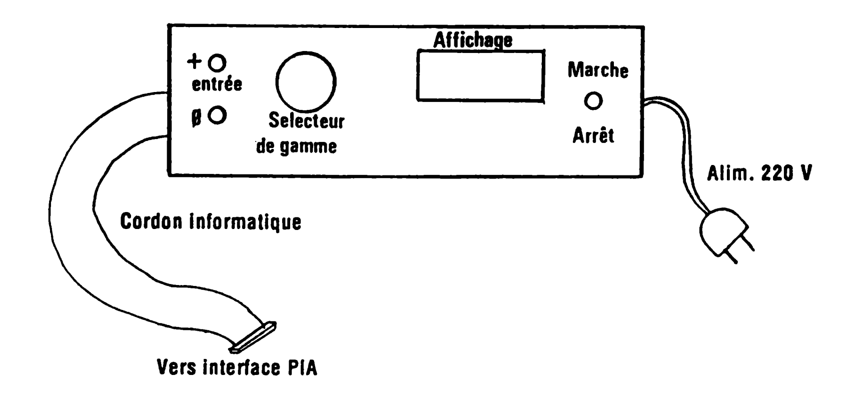


Fig. 16.10

PROGRAMME "DIGI1"

Exemple de programme sous GW BASIC

Faire RUN "DIGI1"

```
10 CLS
20 KEY OFF
30 PORTA=&H310
40 PORTB=&H311
50 REGCONTROL=&H313
60 OUT REGCONTROL,&H82
70 OUT PORTA,1
80 FOR A=1 TO 50:NEXT A
90 OUT PORTA,9:MSD=INP(PORTB)AND 15
100 OUT PORTA,3:LSD=INP(PORTB)AND 15
110 OUT PORTA,4:NSD=INP(PORTB)AND 15
120 OUT PORTA,0
130 AFFICHE=MSD*100+NSD*10+LSD
140 PRINT AFFICHE;
150 GOTO 70
```

EXPLICATION DU PROGRAMME SORTIE, PORT B EN ENTRÉE

ligne 70 : blocage de la mesure,

ligne 80 : temporisation qui peut être augmentée si votre ordinateur a une vitesse d'horloge supérieure à 2 mHz),

ligne 90 : validation de MSD sur le bus de transfert vers l'ordinateur. Transfert de la valeur masquée à 15 dans le registre de l'ordinateur appelé MSD,

ligne 100 : idem pour LSD,

ligne 110 : idem pour NSD avec en plus déblocage de la mesure (90 OUT 8+1 100 OUT 2+1 110 ne bloque plus le HOLD seulement 4),

ligne 120 : bus du digivolt plus sélectionné, entrée débloquée,

ligne 130 : calcul de la valeur entrée en multipliant MSD par 100, NSD par 10 et en cumulant tout à LSD,

ligne 140 : affichage du résultat,

ligne 150 : retour à 70.

Noter que les 4 ms de la prise de mesure sont masquées par le temps de calcul et d'affichage, en cas de micro avec coprocesseur ou tournant à plus de 16 MHz, il faudra rajouter une boucle de temporisation entre 140 et 150.

PROGRAMME "DIGI2"

Programme sous GW BASIC.

Ce programme affiche la mesure en tenant compte de la gamme sélectionnée.

Faire RUN DIGI2

```
5 '___programme__digi2__gammes__
10 CLS
20 KEY OFF
30 PORTA=&H310
40 PORTB=&H311
50 REGCONTROL=&H313
60 OUT REGCONTROL,&H82
70 U=INP(PORTB)AND 48
80 U=U/16
90 OUT PORTA,1
100 FOR A=1 TO 50:NEXT A
110 OUT PORTA,9:MSD=INP(PORTB)AND 15
120 OUT PORTA,3:LSD=INP(PORTB)AND 15
130 OUT PORTA,4:NSD=INP(PORTB)AND 15
140 OUT PORTA,0
150 IF LSD=11 THEN GOTO 220
160 ON U GOSUB 190,200,210
170 LOCATE 10,25:PRINT USING"###.##                         ";AFFICHE
180 GOTO 70
190 AFFICHE=MSD+NSD/10+LSD/100:RETURN
200 AFFICHE=MSD*10+NSD+LSD/10:RETURN
210 AFFICHE=MSD*100+NSD*10+LSD:RETURN
220 LOCATE 10,25:PRINT " dépassement gamme     "
230 GOTO 70
```

Programme identique à DIGI1 avec en plus :

ligne 70 : entrée de la gamme avec masque 48 l'entrée donne 32 pour gamme 0 à 99,9 V, 16 pour la gamme 0 à 9,99 V et 48 pour la gamme 0 à 999 mV,

ligne 80 : division par 16 de façon à obtenir 1 pour la gamme 0 à 9,99 V, 2 pour la gamme 0 à 99,9 V et 3 pour la gamme 0 à 999 mV,

ligne 150 : test si valeur entrée LSD=11 (ce qui correspond à E sur l'affichage du digivolt) soit erreur et saut en 220,

ligne 160 : suivant la gamme saut en 190, 200 ou 210,

ligne 190 :
ligne 200 : } en fonction de la gamme calcul de la mesure puis retour en 170
ligne 210 :

ligne 220 : afficher dépassement de gamme,

ligne 230 : retour en 70.

PROGRAMME ROBOTPC SOUS MS DOS

Composer ROBOTPC puis "entrée" (enter)

On a le choix entre :

1 — Test Entrée/Sortie

2 — Voltmètre géant

3 — Oscilloscope

4 — Retour au DOS

TEST ENTRÉE/SORTIE

Composer 1

Le programme teste en permanence le port B en entrée et affiche les entrées activées. De même, en composant un nombre (si le nombre entré ne fait qu'un ou deux chiffres appuyer sur touche Entrée (enter) pour valider, si le nombre est composé de 3 chiffres, il sera validé automatiquement après composition au clavier du 3e), le programme envoie ce nombre en binaire sur le port A configuré en sortie. Un nombre inférieur à 0 ou supérieur à 255 sera refusé. Appuyer sur F2 pour revenir au choix du programme.

PROGRAMME VOLTGÉANT

Composer 2

Affiche en géant la mesure du digivolt en signalant les éventuels dépassements de gammes. En appuyant sur N, l'affichage ne se fait qu'en normal, sur G en géant.

Retour au choix de programme en appuyant sur F2.

PROGRAMME OSCILLOSCOPE AVEC DIGIVOLT

Composer 3

Le programme enregistre des données à une vitesse de 20 à environ 100 par seconde (suivant type d'ordinateur environ 100 par seconde pour un AT Cadencé à 12 MHz). Il ne fonctionne que sur écran graphique couleur.

Ces données sont ensuite figurées à l'écran sous la forme d'une courbe. L'ordinateur, en même temps que la courbe, peut afficher la mesure en chiffre, mais cela ralentit sa vitesse d'affichage.

Au début du programme, l'ordinateur vous demande si vous voulez l'affichage de la mesure numérique ou non.

Appuyer sur N pour non.

L'ordinateur vous demande de choisir la gamme où vous voulez travailler. Par exemple, si votre tension à afficher ne dépasse pas 10 V, appuyer sur 2 (gamme 0 à 9,99 V).

L'ordinateur en haut de l'écran vous demande :

— Menu=F2 --- Retour au menu, appuyer sur F2

— Gamme=G --- Retour à la sélection de gamme, appuyer sur G

— Mesure=M --- Tracés de courbe permanent, appuyer sur M

(le programme affichera environ 600 points successifs, jusqu'au bord gauche de l'écran, puis effacera la courbe obtenue et en recommencera une nouvelle à la façon d'un oscilloscope. L'arrêt de la courbe sera obtenu en appuyant sur barre espace).

— Mesure avec arrêt=A. Idem au précédent, mais après 600 points affichés à l'écran, en atteignant le bord gauche, la courbe s'arrête. Il faudra appuyer sur A ou M pour recommencer une mesure.

En cas de dépassement de gammes lors de la mesure, l'ordinateur n'affiche pas les points, mais continue à incrémenter l'axe X. Si la mesure revient dans la gamme concernée, l'affichage reprend.

EXEMPLE DE MESURE

Charge du condensateur *(fig.* 16.11)

Sélectionnez le programme en A (mesure avec arrêt) et en même temps, fermez l'interrupteur ou réunissez à l'aide d'un cordon la 2,2 kΩ au + 9 V. La charge du condensateur va se dessiner à l'écran.

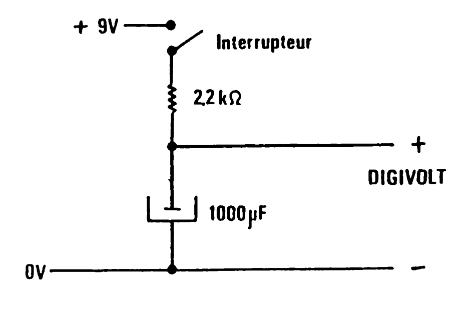


Fig. 16.11

Décharge du condensateur (*fig.* 16.12)

Après avoir chargé le condensateur.

En même temps que vous réunissez la 2,2 kΩ au 0 V, sélectionnez le programme en A, la décharge du condensateur s'affiche à l'écran.

Retour au DOS

Composer 4

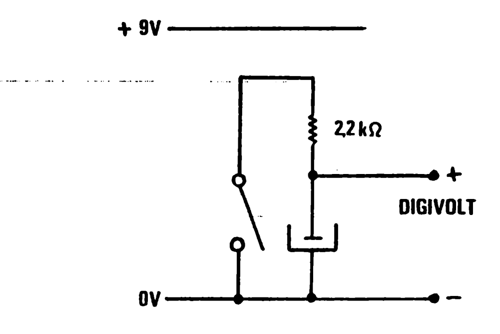


Fig. 16.12

⑰ Moteur pas à pas

PRINCIPE

C'est un moteur possédant un nombre important de positions stables successives séparées entre elles par un Pas, le nombre de Pas définissant le moteur correspond à un tour. Dans notre cas, le moteur est un 48 Pas, c'est-à-dire qu'il faudra 48 impulsions dans une ou plusieurs des bobines du moteur pour exécuter un tour complet.

Chacune des 4 bobines du moteur devra être commandée dans un ordre bien établi, cet ordre définissant le sens de rotation du moteur.

Lors du démarrage du moteur, compte tenu des frictions mécaniques maximum, une commande d'accélération progressive sera nécessaire.

De même, afin d'éviter le dépassement d'un ou de plusieurs Pas de la position désirée lors d'un arrêt brutal par inertie du moteur, il faudra prévoir une décélération progressive plusieurs Pas avant cette position.

Des commandes trop rapides de l'enchaînement des Pas peuvent aboutir à des blocages du moteur. Il faudra donc définir une vitesse maximum du moteur et séparer chaque Pas par une temporisation correspondant à cette vitesse.

TECHNOLOGIE

Cette carte permet la commande d'un moteur Pas à Pas unipolaire par un ordinateur.

Vous pourrez faire tourner le moteur dans un sens ou dans l'autre, le faire avancer ou reculer du nombre de Pas désirés.

CONNEXION

La carte « Moteur Pas à Pas » est connectable à l'interface PIA/PC (disponible en kit ou monté).

L'alimentation de la carte « Moteur Pas à Pas » est externe. Elle doit fournir une tension continue de 12 V (+ ou – 10 %) et débiter un courant minimal de 1A.

SCHÉMA DE PRINCIPE

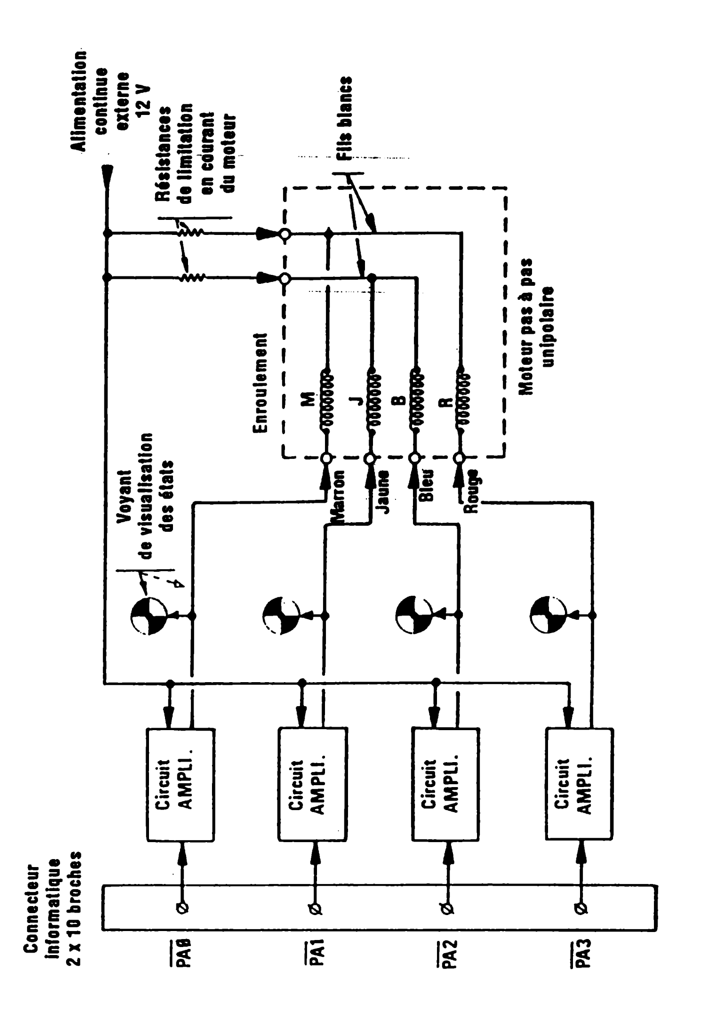


Fig. 17.1

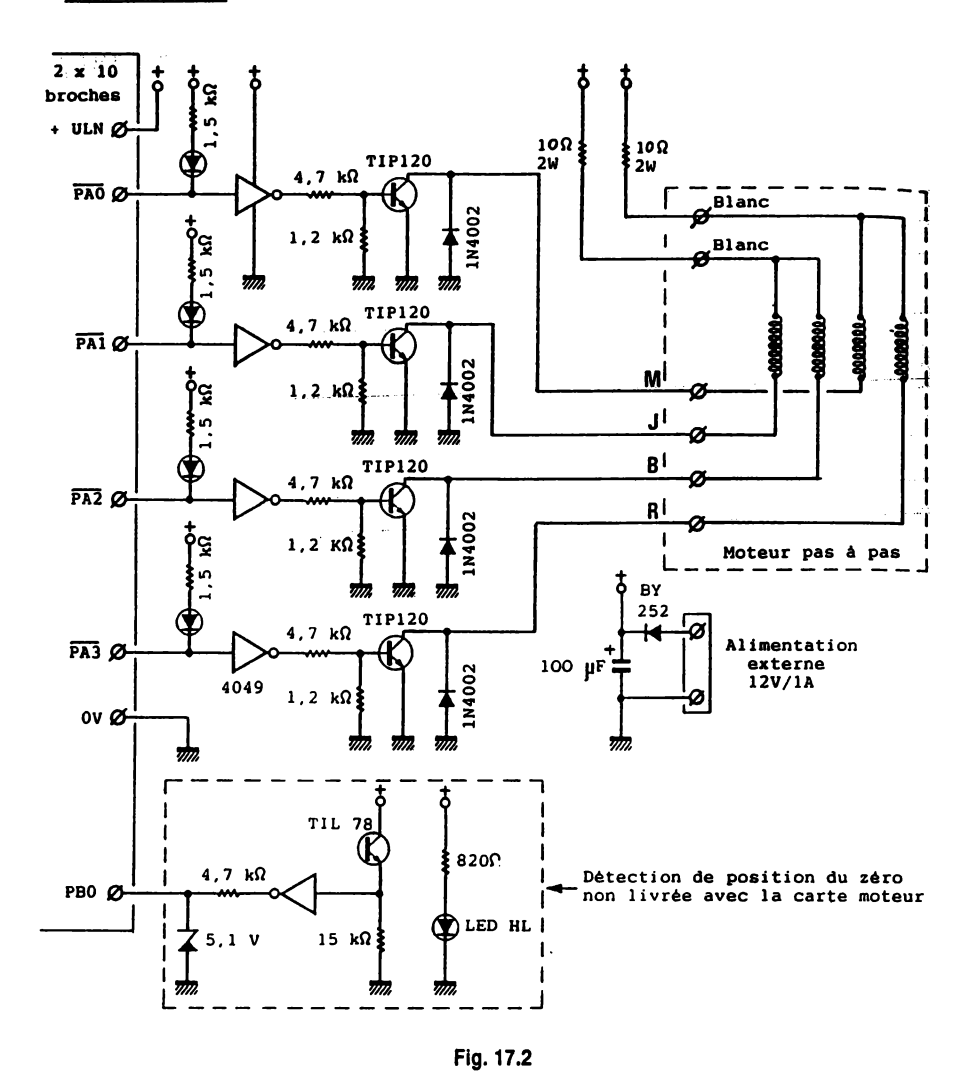


Fig. 17.2

DESCRIPTION DES ÉLÉMENTS

Circuits d'amplifications

Réalisés par un circuit intégré 4049, et des transistors darlingtons, ils permettent la commande des bobines du moteur Pas à Pas.

Les Leds visualisent les bobines alimentées (Led allumée = bobine concernée alimentée).

Moteur Pas à Pas

Type unipolaire

Nombre de Pas par tour : 48 (Pas complet) 96 (par demi-Pas)

Angle de Pas : 7,5° (Pas complet) 3,75° (demi-Pas)

Courant max. par bobine : 300 mA

Principe

Pour faire tourner un moteur Pas à Pas, il faut donc alimenter ses bobines suivant des séquences bien définies.

Il existe deux types de séquences :

— les séquences en Pas complet : on obtient 7,5° par Pas, et il faudra faire 12 fois le tableau de séquences (tableau 1) pour faire un tour complet (ou 48 Pas) du moteur,

— les séquences en demi-Pas : on obtient 3,75° par Pas, et il faudra faire 12 fois le tableau de séquences (tableau 2) pour obtenir un tour complet (ou 96 Pas) du moteur.

TABLEAUX DES SÉQUENCES DE PROGRAMMATION POUR UN MOTEUR PAS À PAS UNIPOLAIRE

Tableau 1

Rotation du moteur par séquence en Pas complet dans le sens anti-horaire (voir programme « Moteur 1 »), angle de pas : 7,5°.

	Enroulement				
Numéros des séquences	**PA3/R**	**PA2/B**	**PA1/J**	**PA0/M**	**Valeur BCD**
1	0	1	0	1	5
2	0	0	1	1	3
3	1	0	1	0	10
4	1	1	0	0	12
1	0	1	0	1	5

retour séquence n° 1

Dans le tableau, les lettres R, B, J, M correspondent aux fils du moteur rouge, bleu, jaune, marron.

Remarque : Pour faire tourner le moteur dans le sens horaire, commencer par la séquence n° 4, puis n° 3, etc...

Tableau 2

Rotation du moteur par séquences en demi-pas dans le sens horaire (voir programme « moteur 3 »), angle de pas : 3,75°.

	Enroulement				
Numéros des séquences	**PA3/R**	**PA2/B**	**PA1/J**	**PA0/M**	**Valeur BCD**
1	1	1	0	0	12
2	1	0	0	0	8
3	1	0	1	0	10
4	0	0	1	0	2
5	0	0	1	1	3
6	0	0	0	1	1
7	0	1	0	1	5
8	0	1	0	0	4
1	1	1	0	0	12

retour séquence n°1

MONTAGE

Souder dans l'ordre suivant et le plus près possible du circuit :

— les résistances,

— les diodes dont le sens est repéré par une bague,

— le condensateur : attention à sa polarité,

— les LED en mettant la patte la plus longue au plus,

— les transistors,

— le circuit intégré repéré par un point ou une encoche,

— le bornier à vis pour CI,

— le connecteur mâle coudé 2 × 10 broches,

— la barrette de connexion mâle six points.

Monter le moteur sur la plaque avec les deux vis de diamètre 4 + écrou. Brancher la prise venant du moteur sur la barrette de connexion (voir le sens d'insertion sur le schéma d'implantation : fils de couleur dans le bon ordre).

Amener une alimentation 12 V/1 A sur le bornier à vis en respectant la polarité.

CONNECTIQUE

Brancher le câble 20 conducteurs venant de votre interface sur le connecteur coudé 2 × 10 broches (attention au détrompeur) (*fig.* 17.3.).

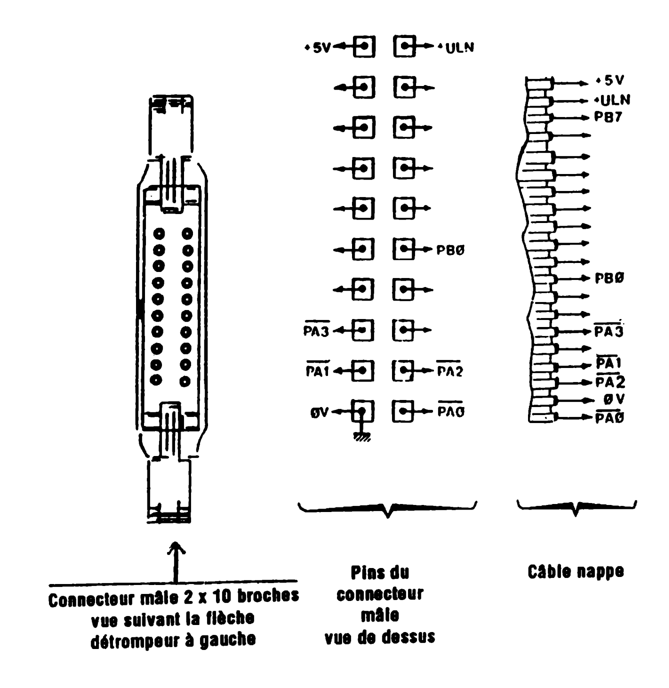


Fig. 17.3

CIRCUIT CÔTÉ CUIVRE

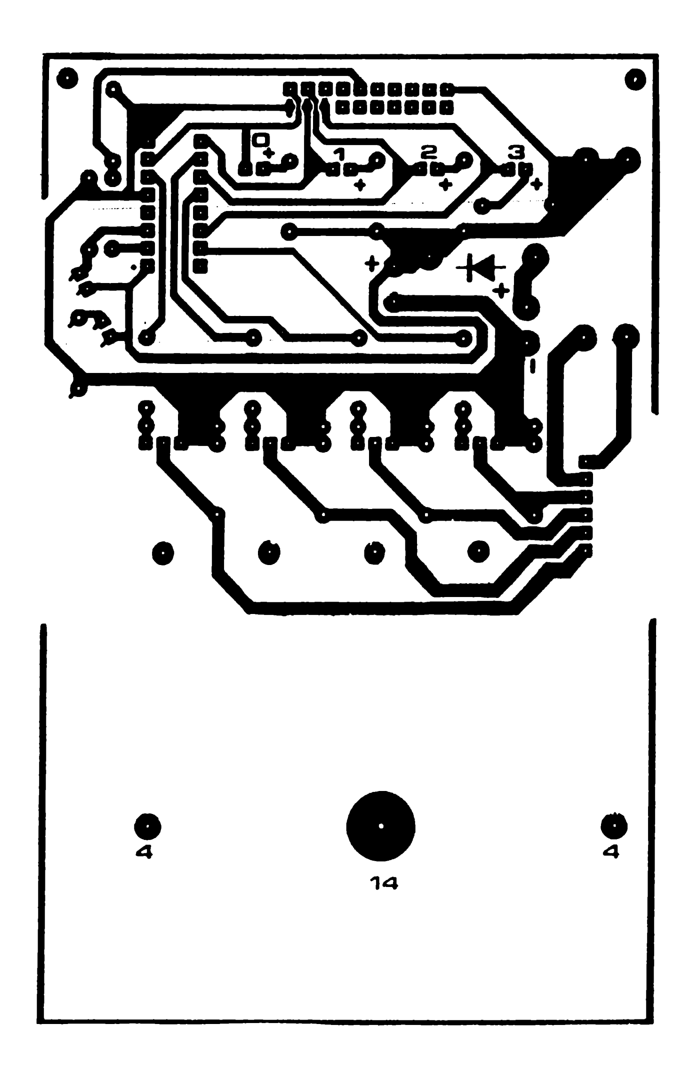


Fig. 17.4

IMPLANTATION

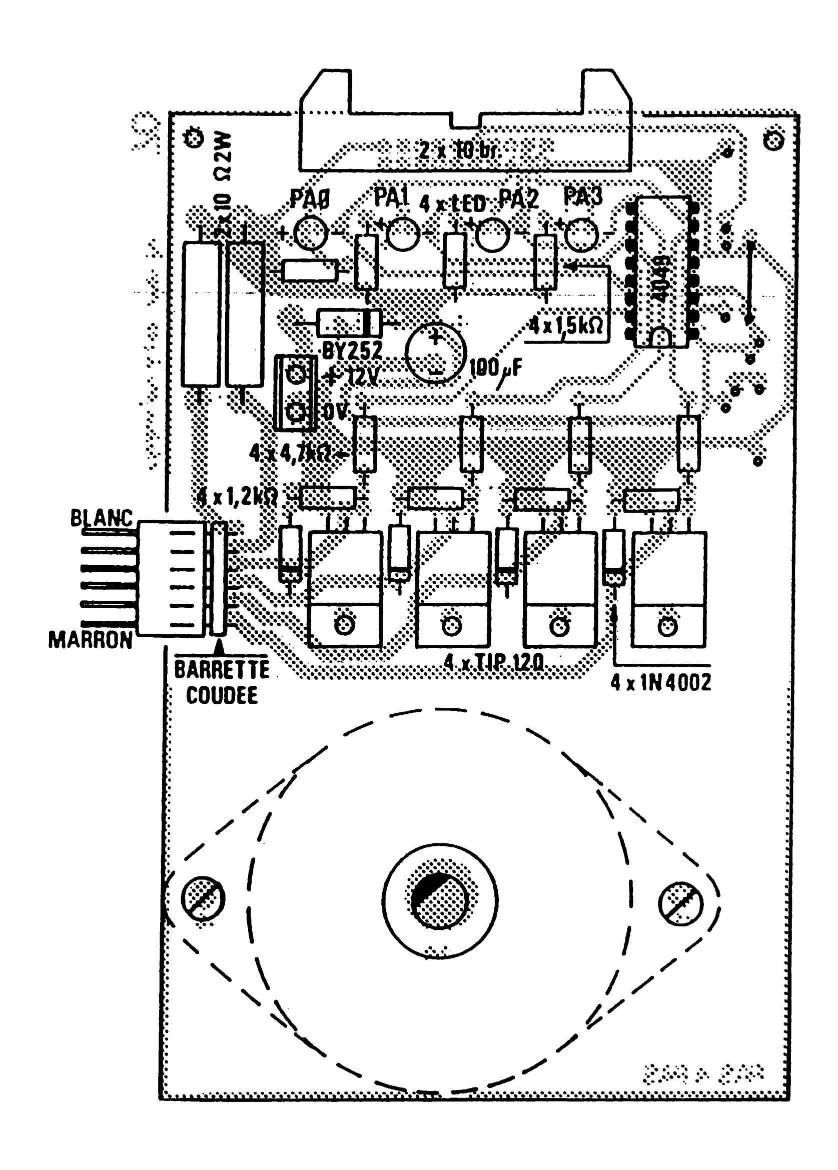
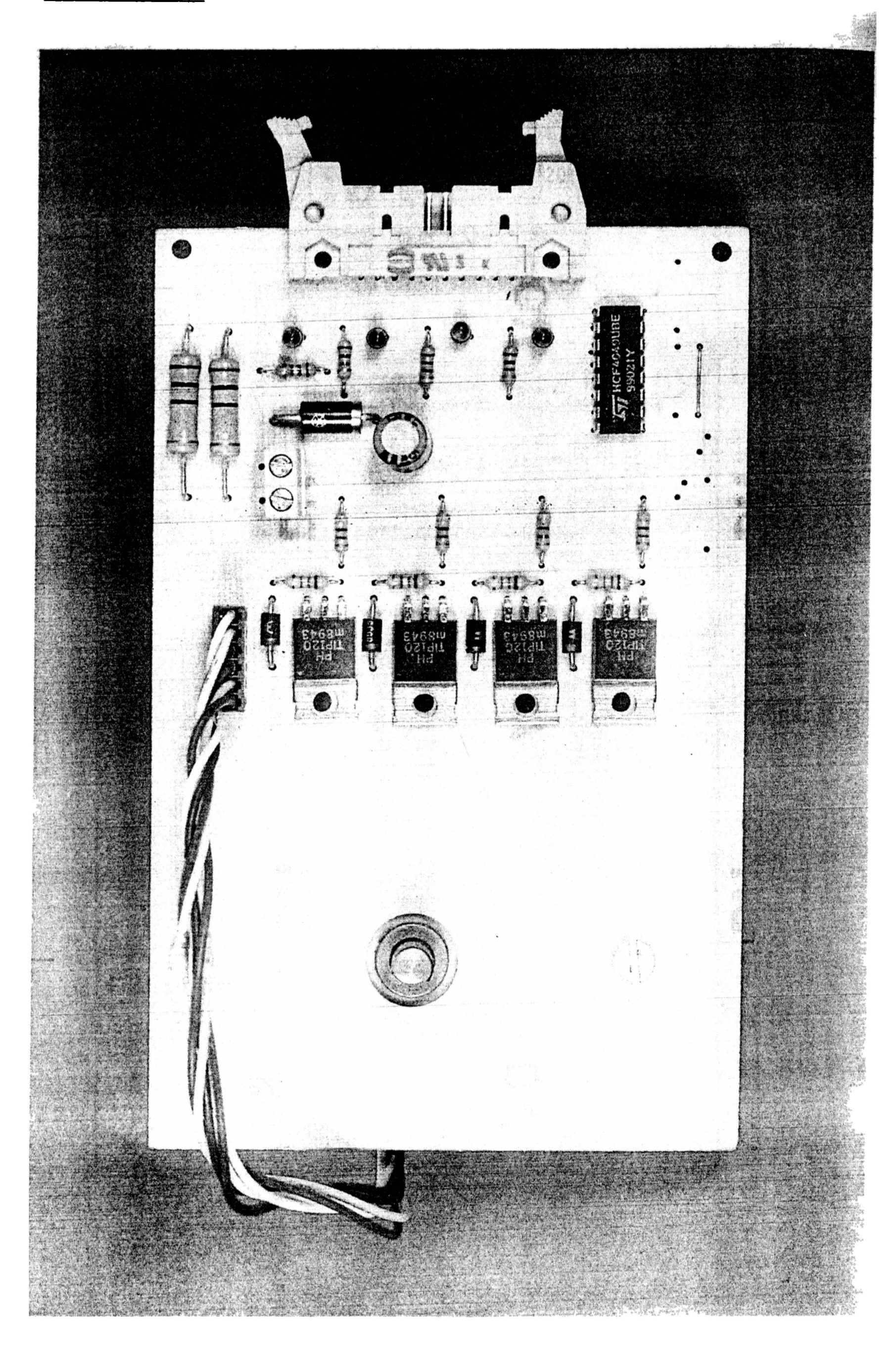


Fig. 17.5


HCF4049UBE
99021Y

MONTAGE DU MOTEUR (fig. 17.6)

Montage du moteur

Barrette de connexion

Vis Ø4

connecteur
2 x 10 broches

Moteur

Circuit coté cuivre

Fig. 17.6

Liste des composants

— 2 × 10 Ω/2 W (marron noir noir),
— 4 × 1,2 kΩ (marron rouge rouge),
— 4 × 1,5 kΩ (marron vert rouge),
— 4 × 4,7 kΩ (jaune violet rouge),
— 1 × 100 µF/16 V ou + radial,
— 4 × 1N 4002 à 4007,
— 1 × BY 252 à 255,
— 4 × TIP 120,
— 4 × LED diamètre 3 rouge,
— 1 × 4049,
— 1 × moteur Pas à Pas 5551-25 DAWC-36 -7,5° par pas,
— 1 × connecteur mâle coudé 2 × 10 broches,
— 1 × barrette de connexion coudée ou droite six broches,
— 1 × bornier à vis deux plots pour CI,
— 2 × vis diamètre 4 × 10 + écrous.

PROGRAMMES

PROGRAMME COMMANDE DE MOTEUR

Exécutable sous BASIC (GW BASIC de Microsoft).

1/ programme "Moteur1" : faire RUN "Moteur1". Le moteur tourne dans le sens inverse des aiguilles d'une montre en 48 Pas

2/ programme "Moteur2" : le moteur tourne dans le sens des aiguilles d'une montre en 48 Pas

3/ programme "Moteur3" : le moteur avance d'un nombre de 1/2 Pas composé au clavier

4/ programme "Moteur4" : après le choix du sens de rotation du moteur à l'aide de + ou –, on peut accélérer ou ralentir la vitesse de rotation du moteur.

PROGRAMME "MOTEUR1"

Ce programme simple permet la commande en Pas complet du moteur Pas à Pas dans le sens anti-horaire.

Une temporisation de valeur constante est réalisée entre chaque Pas par une boucle FOR ... NEXT.

La valeur de cette temporisation (I=20) peut être modifiée selon la fréquence d'horloge de l'ordinateur, ou selon la vitesse de rotation désirée.

Dans le cas d'une temporisation entre chaque Pas de durée trop courte, le moteur se met à vibrer sans tourner. L'action de la touche barre espace permet d'arrêter la rotation du moteur (la touche F2 relance automatiquement le programme).

```
10  CLS:KEY OFF
20  LOCATE 2,15:PRINT "********** Programme moteur pas à pas n°1 **********"
30  LOCATE 6,28:PRINT "Séquences: pas complet"
40  LOCATE 8,30:PRINT "Sens: anti-horaire"
50  LOCATE 20,25:PRINT "(Barr. espace) pour arrêter"
60  REM'-----------initialisation du PIA----------
70  REG.CONTROLE = &H313
80  PORT.A = &H310
90  OUT REG.CONTROLE,&H8B
100 REM'-------Séquences du moteur pas à pas------
110 OUT PORT.A,5:GOSUB 170
120 OUT PORT.A,3:GOSUB 170
130 OUT PORT.A,10:GOSUB 170
140 OUT PORT.A,12:GOSUB 170
150 GOTO 110
160 REM'-----Temporisation entre chaque pas------
170 A$=INKEY$:IF A$=" " THEN END
180 FOR I=1 TO 20:NEXT I
190 RETURN
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : affichage à l'écran des paramètres moteur,

lignes 60 à 90 : initialisation de la carte PIA/PC, définition de l'adresse du registre de contrôle, ainsi que de l'adresse du Port A. Configuration du port A en sortie,

lignes 100 à 150 : elles permettent de charger le port A avec les différentes valeurs BCD selon les séquences en Pas complet du moteur (voir tableau 1). L'instruction GOSUB permet d'aller au sous-programme de temporisation nécessaire entre chaque pas,

ligne 170 : ces instructions permettent de savoir si la touche barre d'espacement a été actionnée et d'arrêter le programme dans ce cas,

lignes 180 à 190 : temporisation, puis retour du sous-programme.

PROGRAMME "MOTEUR2"

Ce programme est comparable au programme précédent, sauf qu'il utilise d'autres instructions pour le déroulement des séquences du moteur Pas à Pas. Il permet la rotation du moteur dans le sens horaire en Pas complet. L'utilisation de l'instruction "DATA" permet de stocker les valeurs (BCD) des différentes séquences du moteur Pas à Pas. L'instruction "READ" permet de lire le contenu de ces données et "RESTORE" initialise les données en début de ligne.

```
10  CLS:KEY OFF
20  LOCATE 2,15:PRINT "********** Programme moteur pas à pas n°2 **********"
30  LOCATE 6,28:PRINT "Séquences: pas complet"
40  LOCATE 8,32:PRINT "Sens: horaire"
50  LOCATE 20,25:PRINT "(Barr. espace) pour arrêter"
60  REM'-----------initialisation du PIA----------
70  REG.CONTROLE = &H313
80  PORT.A = &H310
90  OUT REG.CONTROLE,&H8B
100 REM'-------Séquences du moteur pas à pas------
110 RESTORE 190
120 FOR X=1 TO 4
130 READ PAS(X)
140 OUT PORT.A,PAS(X)
150 FOR I=1 TO 10:NEXT I
160 A$=INKEY$:IF A$=" " THEN END
170 NEXT X
180 GOTO 110
190 DATA 12,10,3,5
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : affichage à l'écran des paramètres moteurs,

lignes 60 à 90 : initialisation de la carte PIA/PC, Port A en sortie,

lignes 100 à 190 : à chaque passage dans la boucle FOR ... NEXT, le programme charge le Port A avec la donnée présente à l'instruction "DATA" (ligne 190) d'indice X. A la fin du chargement de la 4e donnée, le programme va à la ligne 110 initialiser les données au début grâce à l'instruction "RESTORE". La ligne 150 apporte une temporisation entre chaque Pas du moteur permettant ainsi de faire tourner le moteur plus ou moins vite.

PROGRAMME "MOTEUR3"

Ce programme simple permet de faire tourner le moteur du nombre de demi-Pas désiré (1/2 Pas=3,75°) dans le sens horaire. Lors du lancement du programme, le moteur se positionnera à une valeur angulaire d'initialisation (ligne 90). Chaque nouvelle valeur du nombre de demi-Pas souhaité sera effectuée à partir de la position angulaire précédente. Si par exemple vous tapez deux fois de suite le nombre 48, le moteur effectuera un tour complet. Pour arrêter le programme, tapez les touches CTRL + Breack. Pour inverser le sens de rotation du moteur, il suffit d'inverser les

données ligne 210 (voir tableau 2), de changer la valeur d'initialisation angulaire par 12 (ligne 90) et d'autoriser la fonction RESTORE pour la valeur BCD 12 (ligne 180).

```
10  CLS:KEY OFF:RESTORE 210
20  LOCATE 2,15:PRINT "********** Programme moteur pas à pas n°3 **********"
30  LOCATE 6,28:PRINT "Séquences: par demi pas"
40  LOCATE 8,32:PRINT "Sens: horaire"
50  REM'-----------initialisation du PIA----------
60  REG.CONTROLE = &H313
70  PORT.A = &H310
80  OUT REG.CONTROLE,&H8B
90  OUT PORT.A,4
100 REM'-------Chargement du nbr. de demi pas désiré------
110 LOCATE 15,50:PRINT "                              "
120 LOCATE 15,24:INPUT "Nbr. de demi pas désiré ? ",NBR
130 IF NBR <=0 GOTO 110
140 REM'-------Séquences du moteur pas à pas------
150 FOR X=1 TO NBR
160 READ PAS(X MOD 8)
170 OUT PORT.A,PAS(X MOD 8)
180 IF PAS(X MOD 8)=4 THEN RESTORE 210
190 NEXT X
200 GOTO 110
210 DATA 12,8,10,2,3,1,5,4
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : affichage à l'écran des paramètres moteur,

lignes 50 à 80 : initialisation de la carte PIA/PC, Port A en sortie,

ligne 90 : le chargement du Port A, avec la valeur BCD:4, permet d'initialiser la position angulaire du moteur avant d'effectuer un déplacement angulaire,

lignes 100 à 130 : ces instructions permettent d'enregistrer le nombre de demi-pas désiré (variable NBR),

lignes 140 à 210 : le nombre de passage dans la boucle FOR ... NEXT est paramétré par la variable NBR, définissant ainsi le nombre de demi-pas à effectuer.

PROGRAMME "MOTEUR4"

```
10  CLS:KEY OFF
20  LOCATE 2,15:PRINT "********** Programme moteur pas à pas n°4 **********"
30  LOCATE 6,28:PRINT "Séquences: pas complet"
40  TOT=10
50  REM'-----------initialisation du PIA----------
60  REG.CONTROLE = &H313
70  PORT.A = &H310
80  OUT REG.CONTROLE,&H8B
90  '-------Séquences du moteur pas à pas------
100 LOCATE 20,15:PRINT "composer + sens horaire - sens inverse  "
110 LOCATE 21,15:PRINT "        arrêt programme barre espace          "
120 A$=INKEY$
130 IF A$="" GOTO 120
140 IF A$="+" THEN GOTO 180
```

```
150 IF A$="-" THEN GOTO 180
160 IF A$=" " THEN CLS:END
170 BEEP:GOTO 120
180 LOCATE 20,15:PRINT " (Barr. espace) pour arrêter                    "
190 LOCATE 21,15:PRINT " touche + plus vite, touche - moins vite "
200 IF A$="+" THEN GOTO 270
210 '----sens---inverse------
220 FOR X=4 TO 1 STEP -1
230 GOSUB 420
240 NEXT X
250 GOTO 220
260 -----SENS---HORAIRE------
270 FOR X=1 TO 4
280 GOSUB 420
290 NEXT X
300 GOTO 270
310 OUT PORT.A,PAS
320 '--------temporisation------
330 FOR I=1 TO TOT
340 A$=INKEY$
350 IF A$=" " THEN OUT PORT.A,0:GOTO 100
360 IF A$="-" THEN TOT =TOT +5
370 IF TOT<10 THEN GOTO 390
380 IF A$="+" THEN TOT =TOT -5
390 NEXT I
400 RETURN
410 '-------chargement--code----pas----
420 ON X GOTO 430,440,450,460
430 PAS=12:GOTO 310
440 PAS=10:GOTO 310
450 PAS=3:GOTO 310
460 PAS=5:GOTO 310
```

EXPLICATION DU PROGRAMME

lignes 120 à 170 : entrée dans A$ de + ou – selon le sens de rotation désiré, si barre d'espacement retour au basic, sinon bip retour à 130,

lignes 210 à 250 : sens inverse décrémentation de 4 à 1,

lignes 260 à 300 : sens horaire incrémentation de 1 à 4,

lignes 300 à 400 : sortie de la valeur PAS, temporisation de passage d'un pas à l'autre par la boucle NEXT FOR avec augmentation des temps si touche – appuyée. Diminution (limitée à moins de dix boucles) si touche plus appuyée. Si barre d'espacement appuyée, retour à 100 après avoir remis le code PAS à 0 en sortie,

lignes 410 à 460 : chargement dans PAS du code de séquence du moteur en fonction de l'incrémentation ou de la décrémentation dans X.

⑱ Programmateur d'EPROM

PRÉSENTATION

Une EPROM permet de mémoriser une quantité importante de données (sur 8 bits) à de multiples adresses. L'EPROM garde ses informations même après coupure de son alimentation. Seul un rayonnement ultra-violet mis à proximité de la fenêtre de l'EPROM permet d'effacer son contenu (toutes les données passent à 1 après effacement).

La carte programmateur d'EPROM permet donc de lire le contenu d'une EPROM ou de charger des données dans sa mémoire.

TECHNOLOGIE

Le programmateur d'EPROM est constitué autour d'une logique qui permet la lecture ou la programmation selon l'algorithme standard de programmation de différents types d'EPROM : 2764, 27128, 27256 et 27512. (EPROM 1M Bits avec adaptateur).

Un commutateur permet également de sélectionner les différentes tensions de programmation (V_{PP}) : 5 V, 12, 5 V, 21 V et 25 V (selon fabriquant). Un voyant, signale à l'utilisateur qu'il ne faut pas retirer l'EPROM du support sous risque de détériorer celle-ci, car on se trouve en mode écriture.

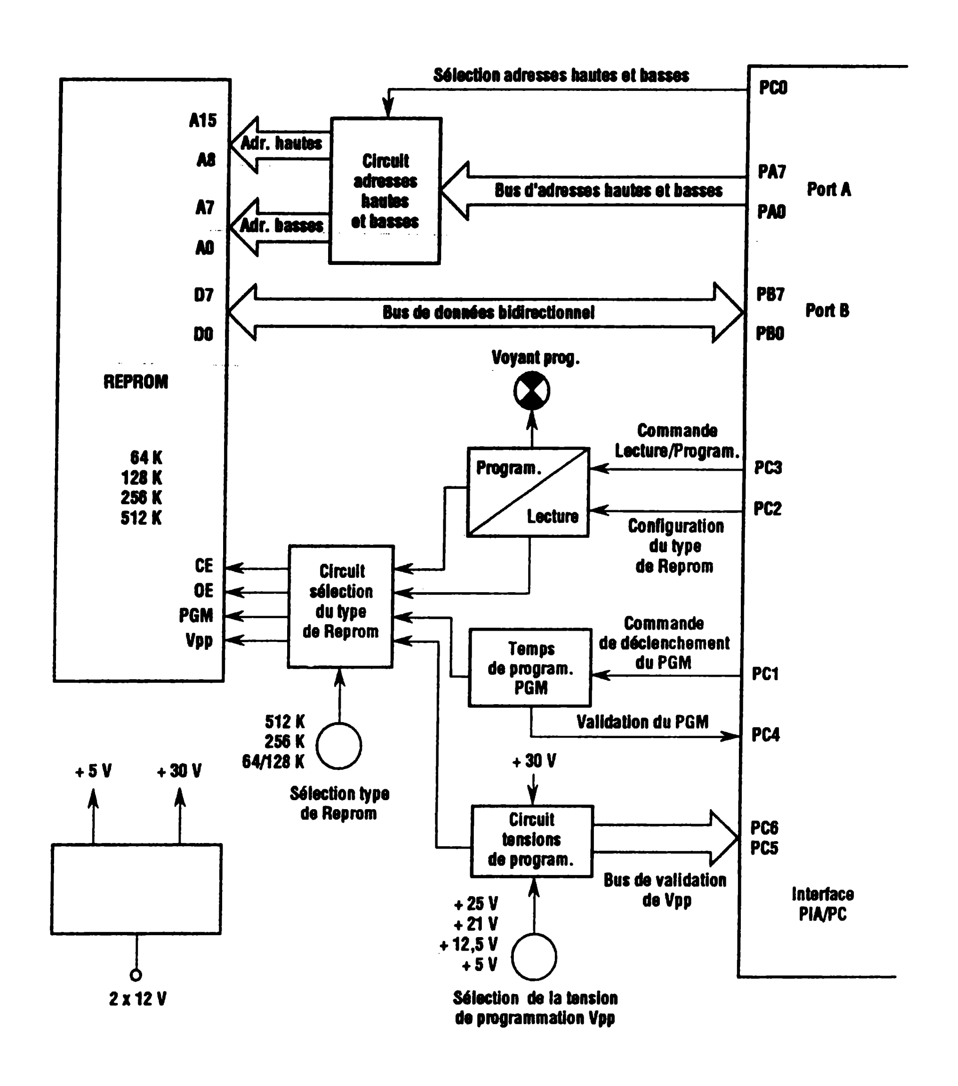
SCHÉMAS (*fig.* 18.1 et 18.2)

COMPATIBILITÉ ET CONNECTIONS

Le programmateur peut être relié à tout ordinateur compatible PC : XT/AT par l'intermédiaire de la carte PIA/PC (disponible en kit ou montée).

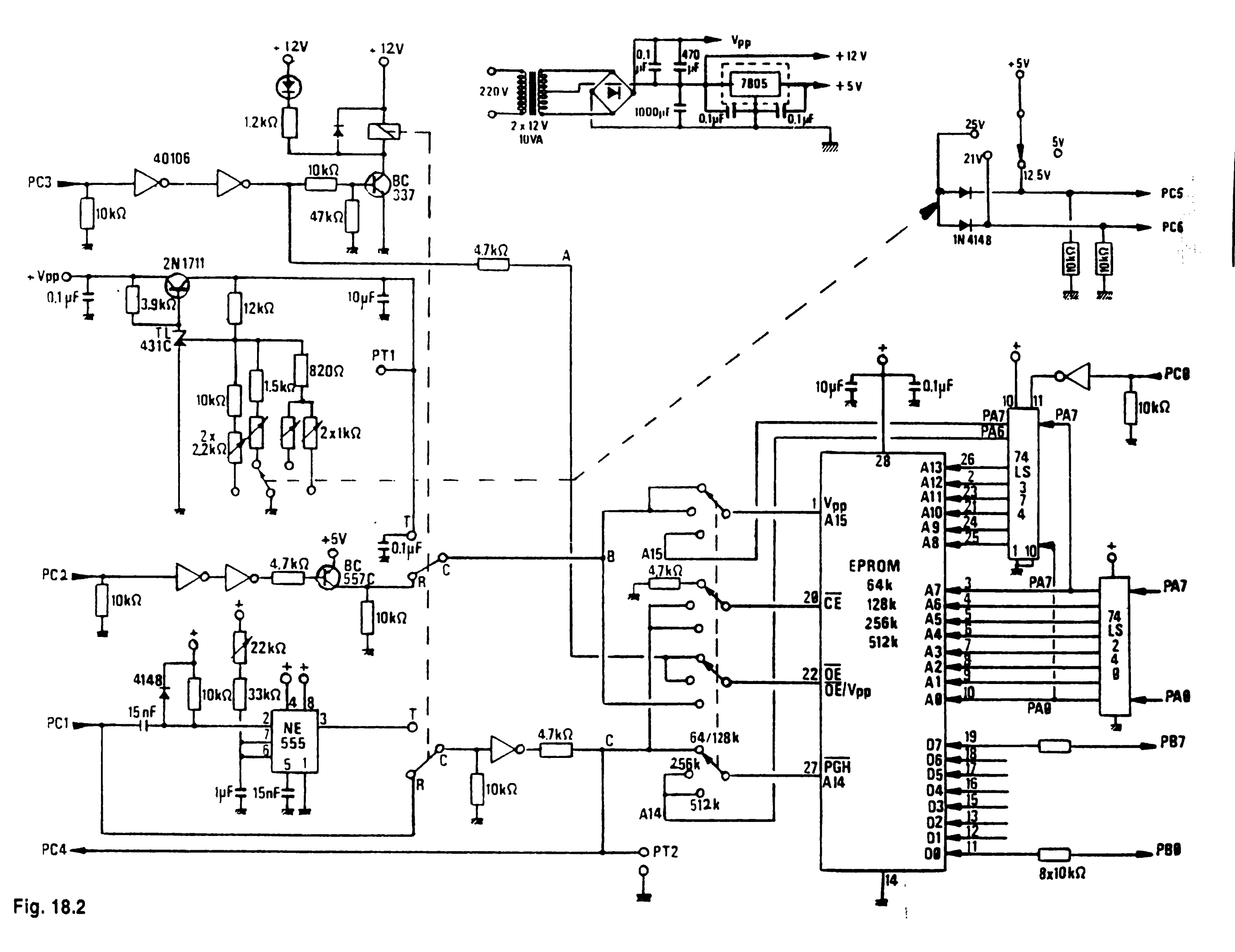
Deux câbles nappe avec prises 20 broches et 10 broches permettent de relier le programmateur à la carte PIA/PC.

L'alimentation externe du programmateur est réalisée par un transformateur 2 × 12 V/10 VA.



Sélection adresses hautes et basses
PC0
A15
A8
Adr. hautes
Circuit adresses hautes et basses
PA7
Bus d'adresses hautes et basses
Port A
PA0
A7
Adr. basses
A0
D7
PB7
Bus de données bidirectionnel
Port B
D0
PB0
Voyant prog.
REPROM
64 K
128 K
256 K
512 K
Commande Lecture/Program.
Program.
PC3
Lecture
PC2
Configuration du type de Reprom
CE
OE
PGM
Vpp
Circuit sélection du type de Reprom
Commande de déclenchement du PGM
Temps de program. PGM
PC1
Validation du PGM
PC4
512 K
256 K
64/128 K
Sélection type de Reprom
+ 5 V
+ 30 V
+ 30 V
Circuit tensions de program.
PC6
PC5
Bus de validation de Vpp
Interface PIA/PC
+ 25 V
+ 21 V
+ 12,5 V
+ 5 V
2 x 12 V
Sélection de la tension de programmation Vpp

Fig. 18.1



PC3
40106
10kΩ
BC 337
47kΩ
1.2kΩ
+12V
+Vpp
2N1711
3.9kΩ
TL 431C
12kΩ
820Ω
1.5kΩ
2x 2.2kΩ
2x1kΩ
PT1
PC2
4.7kΩ
BC 557C
+5V
PC1
4148
15nF
22kΩ
33kΩ
NE 555
1µF
PC4
PT2
220V
2 x 12 V 10VA
1000µF
7805
Vpp
+12 V
+5 V
25V
21V
12.5V
5V
1N4148
PC5
PC6
PC8
74 LS 374
74 LS 244
PA7
PA6
PA0
EPROM 64k 128k 256k 512k
Vpp A15
CE
OE/Vpp
PGH A14
64/128k
256k
512k
A15
A14
PB7
PB0
8x10kΩ

Fig. 18.2

CARACTÉRISTIQUES ET PRINCIPES DE LECTURE OU DE PROGRAMMATION D'UNE EPROM

Caractéristiques d'une EPROM :

Une EPROM de type 2764 ... 512 se caractérise par une donnée sur huit bits (un octet) à chaque adresse sélectionnée.

	2764	27128	27256	27512
Nombre de kilo bits disponible	64 k bits	128 k bits	256 k bits	512 k bits
Nombre de kilo octets disponible	8 k octets	16 k octets	32 k octets	64 k octets
Nombre d'adresses disponibles pour un octet de données	8 192 adrs disponibles	16 384 adrs disponibles	32 768 adrs disponibles	65 536 adrs disponibles
Bits du Bus d'adresse de A_0 ... A_n	de A_0 à A_{12}	de A_0 à A_{13}	de A_0 à A_{14}	de A_0 à A_{15}

Remarque :

1 KBits = 1 024 bits, 1 Koctets = 1 024 octets = 8 192 bit.

Principe de lecture d'une EPROM

Pour lire le contenu d'un octet à une adresse définie, il faut présenter l'adresse souhaitée sur son bus d'adresse, puis lire les données présentes sur son bus de données après avoir sélectionné les entrées CE (validation de l'EPROM) et OE (validation du bus de donnée) aux niveaux bas.

Principe de programmation d'une EPROM

Une EPROM de type 2764 ... 512 est programmable électriquement, c'est-à-dire qu'il faut lui fournir une tension de programmation (V_{pp}) supérieure à sa tension d'alimentation, pendant un temps de 50 ms, afin qu'elle puisse mémoriser un mot de 8 Bits présent sur son bus de données à l'adresse désirée.

La pin marquée V_{pp} permet de lui appliquer la tension de programmation. La pin PGM (actif au niveau bas) permet de valider le temps de programmation pendant 50 ms.

PRINCIPE DE FONCTIONNEMENT DU PROGRAMMATEUR D'EPROM

Circuit adresses hautes et basses

Il est constitué par le circuit intégré 74 LS 374, qui permet de latcher les adresses hautes de l'EPROM.

Quand l'adresse haute (A_8 ... A_{15}) de l'EPROM est présente sur le Port A, le passage à l'état bas de PC0 mémorise le contenu du Port A sur les bits d'adresses hautes de l'EPROM. Il ne reste plus qu'à présenter l'adresse basse sur le Port A pour obtenir l'adressage complet de l'EPROM.

Circuit sélection du type d'EPROM

Un commutateur à trois positions permet d'adapter le montage aux différents brochages des EPROMS.

Type de REPROM	Broche 1	Broche 20	Broche 22	Broche 27
2764-128	V_{pp}	CE	OE	PGM
27256	V_{pp}	CE	OE	A_{14}
27512	A_{15}	CE	OE/V_{pp}	A_{14}

Programmation/Lecture

Après avoir sélectionné le type d'EPROM et inséré celle-ci dans le support, un niveau logique haut sur le bit PC3 permet de valider la tension de programmation et le temps de programmation PGM. Un voyant rouge indique qu'il ne faut pas retirer l'EPROM de son support (risque de détérioration).

Temps de programmation PGM

Pour plus de fiabilité et d'adaptabilité aux différents ordinateurs, la temporisation de 50 ms (plus ou moins 5 ms) est réalisée par un circuit intégré NE 555.

Cette temporisation est déclenchée par un front descendant sur le bit PC1 de la carte PIA/PC. Le logiciel de programmation est informé de la fin de cette temporisation grâce au bit PC4 et attendra avant d'exécuter d'autres instructions.

Tension de programmations

Une zéner programmable (TL 431) permet d'obtenir différentes tensions ajustables.

La position du commutateur est recopiée sur les bits PC5 et PC6 de la carte PIA/PC. Après réglage et sélection des tensions de programmation, le logiciel connaîtra la tension sélectionnée par l'utilisateur.

PC6	PC5	Valeur en BCD de PC5 et PC6	Tension sélectionnée après réglage
0	0	0	5 V
0	1	32	12,5 V
1	0	64	21 V
1	1	96	25 V

RÉGLAGE DE LA CARTE EPROM

Réglage tension V_{pp} (Pt1)

Connecter un voltmètre (calibre supérieur à 25 V) aux points tests marqués "Masse" et "Pt1". Ne pas relier le programmateur à la carte PIA/PC. Sélectionner la tension de programmation la plus basse, puis alimenter le montage. Régler l'ajustable marqué "5 V" jusqu'à obtenir la tension désirée.

Sélectionner les autres tensions de programmation en réglant les ajustables concernées.

Réglage du temps de programmation avec oscilloscope (Pt2)

Connecter la masse de l'oscilloscope au point test marqué "Masse", et brancher la sonde au point test "Pt2".

Raccorder la carte EPROM à l'ordinateur en vérifiant que toutes les alimentations soient débranchées. Alimenter le programmateur, puis l'ordinateur (vérifier qu'il n'y a pas d'EPROM dans le support).

Lancer le programme REG-PGM.BAS écrit en GW BASIC de Microsoft et régler l'ajustable marqué "PGM" jusqu'à obtenir un temps de 50 ms (plus ou moins 5 ms) entre le front descendant et le front montant du signal (*fig.* 18.3).

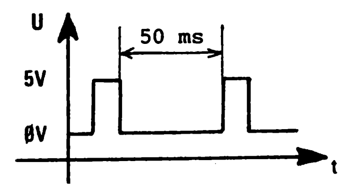
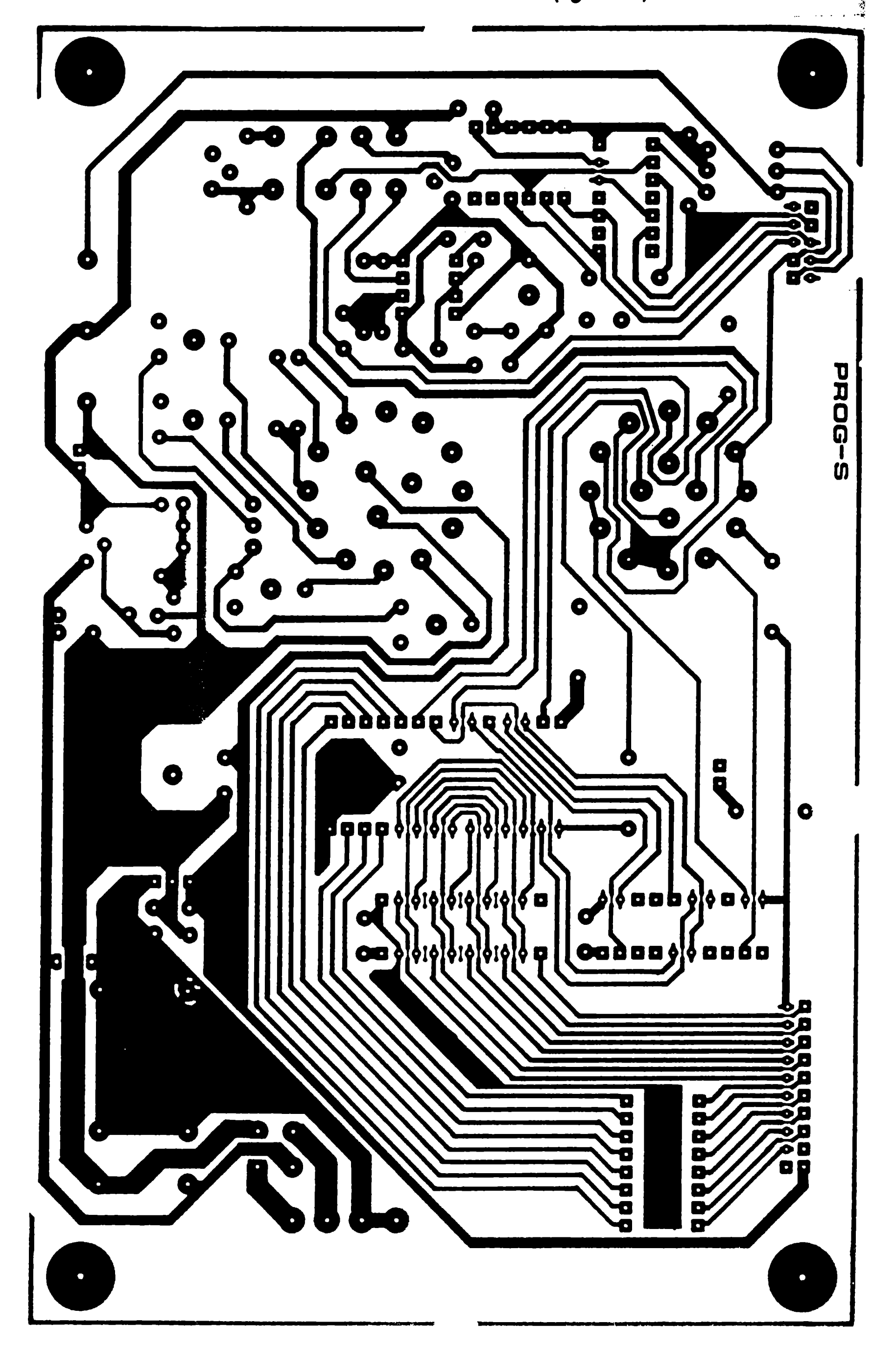
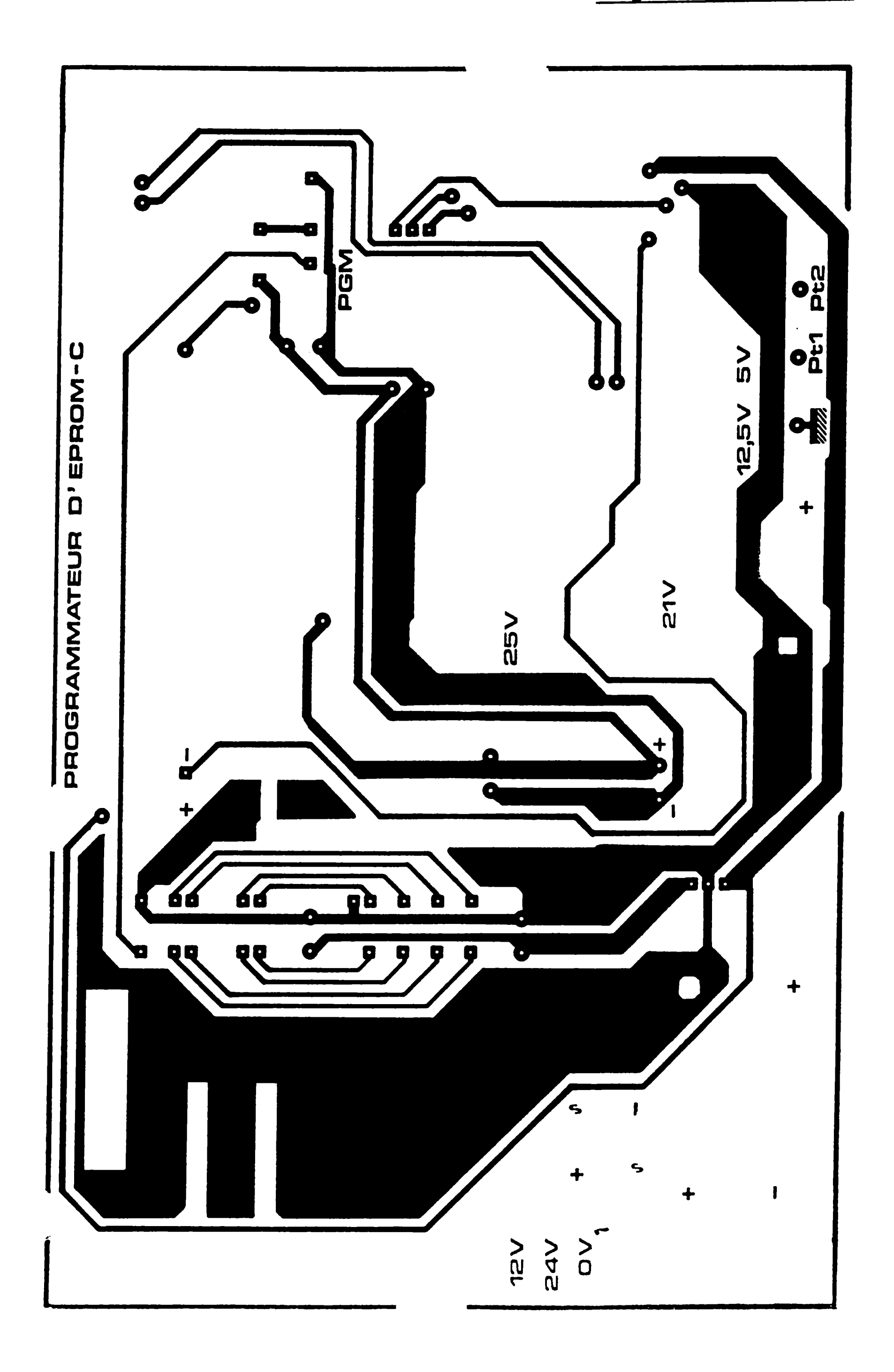


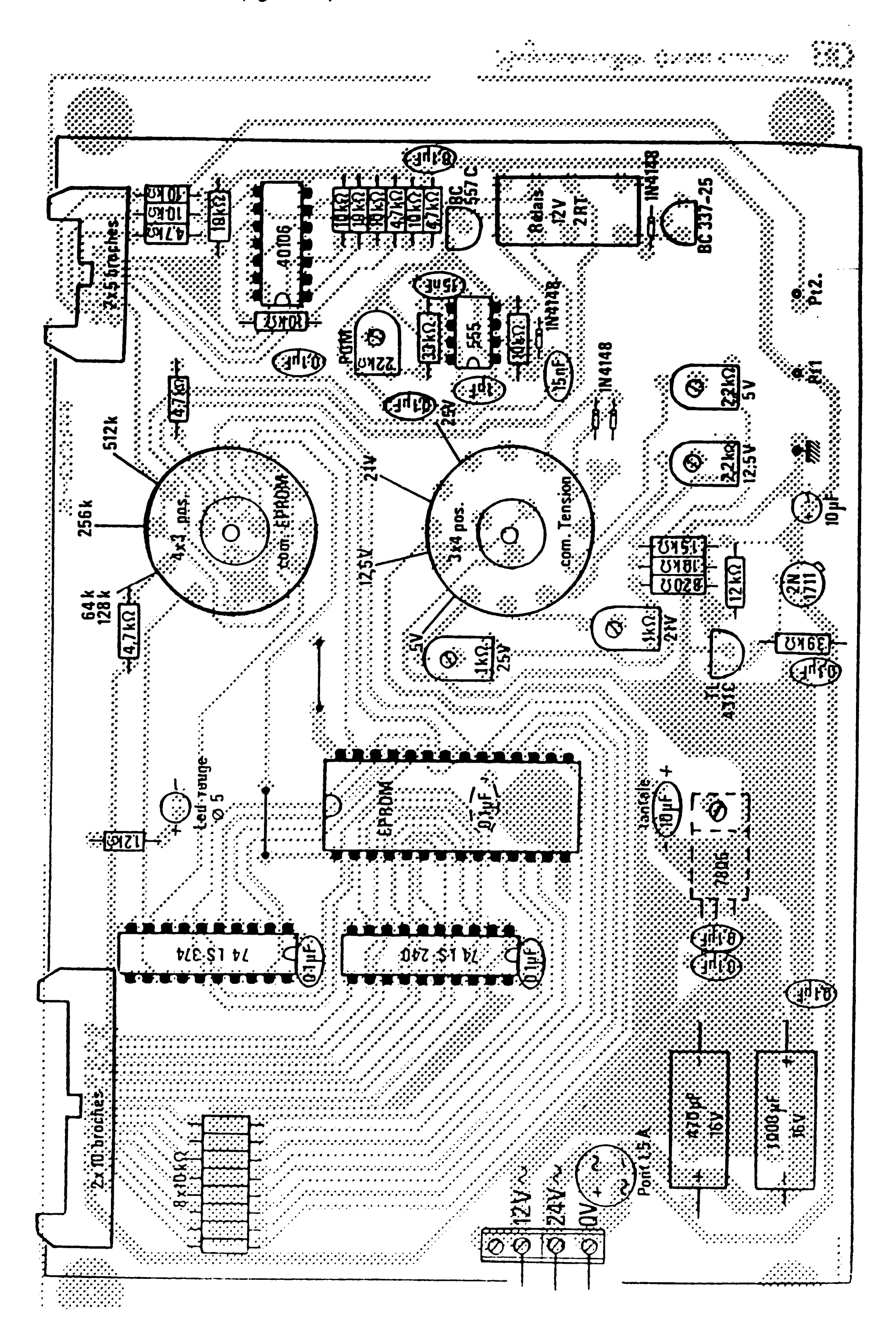
Fig. 18.3

CIRCUIT IMPRIMÉ DE LA CARTE PRINCIPALE (*fig.* 18.4)




PROGRAMMATEUR D' EPROM-C
PGM
25V
21V
12,5V
5V
Pt1
Pt2
12V
24V
0V

IMPLANTATION (*fig.* 18.5)



MONTAGE DE LA CARTE PRINCIPALE

Le circuit étant réalisé en double face, certaines pattes de composants sont à souder côté cuivre et côté composants du circuit :

Souder dans l'ordre suivant et le plus près possible du circuit :

— les deux straps repérés •—• sur l'implantation,

— les diodes 1N 4148 (attention au sens indiqué par une bague),

— les résistances,

— les ajustables,

— les condensateurs (attention à la polarité des shimiques et du tantale). Le 0,1 µF dessiné en pointillé au centre de l'EPROM doit être soudé sur le côté cuivre,

— le pont de diodes (voir le sens sur l'implantation),

— la LED rouge qui est polarisée (patte longue au plus),

— les transistors qui s'insèrent dans les trois trous placés en triangle (attention à la position de l'ergot du 2N 1711),

— le TL 431 C,

— le relais,

— les circuits intégrés dont le sens d'insertion est donné par un point ou une encoche,

— les deux commutateurs,

— les deux connecteurs coudés,

— le support à insertion nulle en plaçant la manette du côté du 74 LS 240,

— le 7805 qui est soudé sur le côté cuivre du circuit.

LISTE DES COMPOSANTS

Résistances :

1 × 820 Ω (gris rouge marron),
1 × 1,2 kΩ (marron rouge rouge),
1 × 1,5 kΩ (marron vert rouge),
1 × 3,9 kΩ (orange blanc rouge),
5 × 4,7 kΩ (jaune violet rouge),
18 × 10 kΩ (marron noir orange),
1 × 12 kΩ (marron rouge orange),
1 × 33 kΩ (orange orange orange).

Ajustables horizontaux :

2 × 1 kΩ,
2 × 2,2 kΩ,
1 × 22 kΩ.

Condensateurs :

2 × 15 nF,
10 × 0,1 µF,
1 × 1 µF,
1 × 10 µF radial,
1 × 10 µF/16 V TG,
1 × 470 µF/16 V axial,
1 × 1 000 µF/16 V axial.

Semiconducteurs :

1 × 74 LS 374,
1 × 74 LS 240,
1 × 40106 (ou 4069),
1 × NE 555,
1 × TLC 431,
1 × BC 337-25,
1 × BC 557 C,
4 × 1N 4148,
1 × 2N 1711,
1 × 7805,
1 × pont 1,5 A.

Divers :

1 × commutateur 3 × 4 positions,
1 × commutateur 4 × 3 positions,
1 × relais 12 V (iskra),
1 × LED diamètre 5 rouge,
1 × support à insertion nulle,
1 × transformateur 2 × 12 V/10 VA,
1 × radiateur U,
1 × connecteur coudé 2 × 5 broches,
1 × connecteur coudé 2 × 10 broches,
1 × porte fusible CI,
1 × fusible 100 mA,
6 × bornier à vis pour CI.

Visserie :

1 × vis 3 × 15,
2 × écrou 3.

MONTAGE DU SUPPORT INSERTION NULLE (*fig.* 18.6)

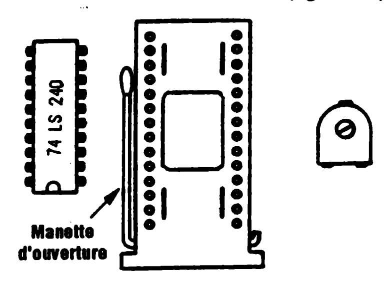


Fig. 18.6

MONTAGE DU 7805 (*fig.* 18.7)

Mettre en place la vis 3 x 15 sur le circuit par le côté composants et la bloquer avec un écrou.

Enfiler ensuite le radiateur, le 7805 après avoir plié ses pattes pour qu'elles entrent dans les pastilles prévues pour les souder. Bloquer avec le dernier écrou et souder les trois pattes sur le côté cuivre et le côté composants.

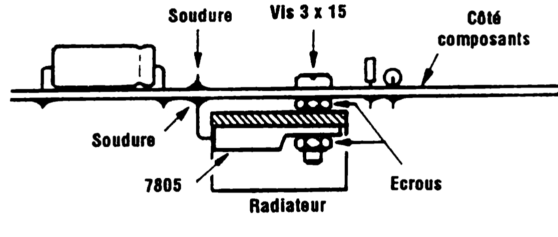


Fig. 18.7

CIRCUIT IMPRIMÉ DE LA CARTE D'ALIMENTATION

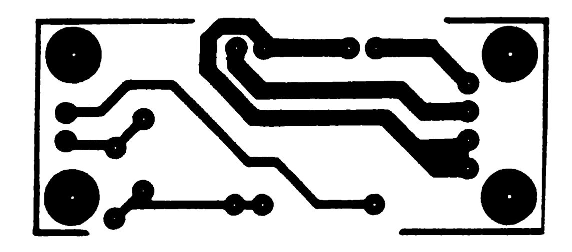


Fig. 18.8

IMPLANTATION DE LA CARTE D'ALIMENTATION

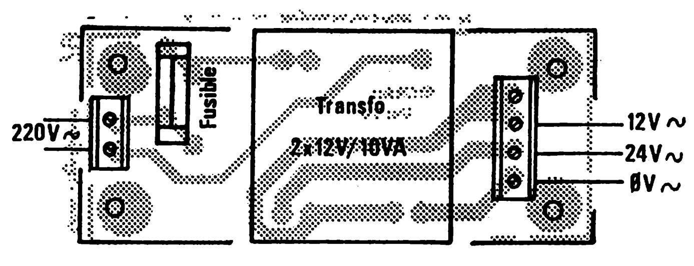


Fig. 18.9

MONTAGE DE LA CARTE ALIMENTATION

Souder le porte fusible, les trois borniers et le transformateur. Mettre en place le fusible dans le porte fusible.

CÂBLAGE

Câbler un fil secteur sur le bornier marqué 220V.

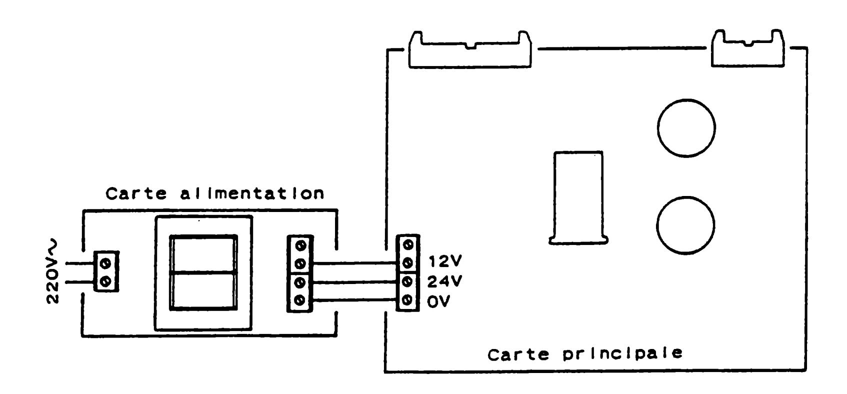


Fig. 18.10

Avec trois longueurs de fil isolé, relier entre eux les borniers de la carte alimentation et de la carte principale marqués de la même manière : 12 V avec 12 V, 24 V avec 24 V et 0 V avec 0 V (*fig.* 18.10).



PROGRAMMES

PROGRAMME "REG-PGM"

Ce programme simple sert au réglage à l'aide d'un oscilloscope de l'ajustable "PGM" (voir réglage).

Il provoque un front descendant sur le bit PC1 déclenchant ainsi le monostable, et attend que cette temporisation soit écoulée pour renouveler perpétuellement ce signal.

```
10  REM -------------------- "REG-PGM " --------------------
20  CLS:KEY OFF
30  PORT.C=&H312
40  REG.CON=&H313
50  MODE.LECTURE=&H8A
60  OUT REG.CON,MODE.LECTURE
70  LOCATE 2,8:PRINT"*** Réglage du temps de programation (Pgm) avec oscilloscop
e ***"
80  LOCATE 4,16:PRINT"(Pgm) : 50 ms de front descendant à front montant"
90  LOCATE 10,22:PRINT"NE PAS METTRE d'EPROM DANS LE SUPPORT!."
100 LOCATE 18,27:PRINT"(Barr. espace) pour arréter"
110 OUT PORT.C,10:OUT PORT.C,8
120 IF (INP(PORT.C) AND 16)=0 THEN 120
130 SP$=INKEY$ : IF SP$<>" " THEN GOTO 110
140 OUT REG.CON,MODE.LECTURE
150 END
```

PROGRAMME "ROMVIERG.BAS"

Ce programme écrit en GW BASIC de Microsoft permet de tester la virginité d'une EPROM (niveau haut sur toutes les données).

```
10  REM -------------------- "ROMVIERG" --------------------
20  CLS:KEY OFF
30  LOCATE 10,20:PRINT"Ne Pas Mettre d'Eprom dans le Support.":GOSUB 290
40  PORT.A=&H310:PORT.B=&H311:PORT.C=&H312
50  REG.CON=&H313:MODE.LECTURE=&H8A
60  CLS:LOCATE 1,14:PRINT"***    Test de Virginité d'une Eprom.     ***"
70  LOCATE 10,24:PRINT"<1>...Eprom de 64 Ko."
80  LOCATE 11,24:PRINT"<2>...Eprom de 128 Ko."
90  LOCATE 12,24:PRINT"<3>...Eprom de 256 Ko."
100 LOCATE 13,24:PRINT"<4>...Eprom de 512 Ko."
110 LOCATE 17,30:INPUT"Choix : ",TEST
120 IF TEST=1 THEN X=0:LONGR=8191:GOTO 160
130 IF TEST=2 THEN X=0:LONGR=16383:GOTO 160
140 IF TEST=3 THEN X=2:LONGR=32767:GOTO 160
150 IF TEST=4 THEN X=6:LONGR=65535!:GOTO 160:ELSE 110
160 OUT REG.CON,MODE.LECTURE:OUT PORT.C,X
170 CLS:LOCATE 10,12:PRINT"Mettre le Commutateur sur le type d'Eprom selectionné
e.":GOSUB 290
```

```
180 CLS:LOCATE 10,23:PRINT"Insérer l'Eprom dans son Support.":GOSUB 290
190 CLS:LOCATE 10,24:PRINT"Test de Virginité en Cours..."
200 FOR ADR=0 TO LONGR
210     ADR.H=INT(ADR/256):ADR.B=ADR-(ADR.H*256)
220     OUT PORT.A,ADR.H
230     OUT PORT.C,X+1:OUT PORT.C,X
240     OUT PORT.A,ADR.B
250     IF INP(PORT.B)<>255 THEN GOTO 280
260 NEXT ADR
270 CLS:BEEP:LOCATE 12,24:PRINT"L'Eprom est Vierge!.":END
280 CLS:BEEP:LOCATE 12,24:PRINT"L'Eprom n'est pas Vierge!.":END
290 LOCATE 12,24:PRINT"<BARRE ESPACE> Une fois Prêt..."
300 SP$=INKEY$:IF SP$<>" " THEN GOTO 300
310 RETURN
```

EXPLICATION DU PROGRAMME

lignes 10 à 30 : effacement et affichage de messages à l'écran,

lignes 40 à 60 : valeurs de configuration du PIA pour le mode de lecture port A en sortie, port B en entrée, demi-port C inférieur en sortie et demi-port C supérieur en entrée,

lignes 70 à 150 : ces instructions permettent d'enregistrer une valeur de configuration (X) et une variable (LONGR) en fonction du type d'EPROM sélectionné,

ligne 160 : configuration de la carte PIA/PC et chargement du port C avec la valeur (X) de configuration,

lignes 170 à 190 : ces lignes permettent d'afficher des messages lors de la manipulation du programmateur,

ligne 200 : utilisation d'une boucle For ... Next paramétrée par la variable "LONGR" permettant d'explorer toutes les adresses de l'EPROM sélectionnées au préalable,

ligne 210 : calcul de l'adresse haute et de l'adresse basse en fonction de l'adresse complète (ADR),

lignes 220 à 240 : chargement de l'adresse haute sur le port A, création d'un front descendant sur PC0 permettant de mémoriser l'adresse haute dans le latche,

lignes 250 à 310 : vérification des données de l'EPROM et retour au début de la boucle avec incrémentation de l'adresse (ADR) si toutes les données sont au niveau haut (EPROM vierge).

PROGRAMME "ROMCOPIE.BAS"

Ce programme permet de dupliquer le contenu d'une EPROM dans une autre EPROM vierge de même type.

On peut également sélectionner différentes tensions de programmation à déterminer selon le fabricant de l'EPROM.

Remarque : Lors de la programmation, ne pas retirer ou insérer une EPROM dans le support tant que la LED de visualisation est allumée.

```
10  REM -------------------- "ROMCOPIE" --------------------
20  CLS:KEY OFF:DEF SEG=&H8000
30  LOCATE 10,24:PRINT"Ne Pas Mettre d'Eprom dans le Support.":GOSUB 500
40  PORT.A=&H310:PORT.B=&H311:PORT.C=&H312
50  REG.CON=&H313:MODE.LECTURE=&H8A:MODE.ECRITURE=&H88
60  CLS:LOCATE 1,14:PRINT"***    Duplication d'une Eprom.    ***"
70  LOCATE 10,24:PRINT"<1>...Eprom de 64 Ko."
80  LOCATE 11,24:PRINT"<2>...Eprom de 128 Ko."
90  LOCATE 12,24:PRINT"<3>...Eprom de 256 Ko."
100 LOCATE 13,24:PRINT"<4>...Eprom de 512 Ko."
110 LOCATE 17,30:INPUT"Choix : ",TEST
120 IF TEST=1 THEN X=0:LONGR=8191:GOTO 160
130 IF TEST=2 THEN X=0:LONGR=16383:GOTO 160
140 IF TEST=3 THEN X=2:LONGR=32767:GOTO 160
150 IF TEST=4 THEN X=6:LONGR=65535!:GOTO 160 ELSE GOTO 110
160 OUT REG.CON,MODE.LECTURE:OUT PORT.C,X
170 CLS:LOCATE 10,12:PRINT"Mettre le Commutateur sur le type d'Eprom selectionné
e.":GOSUB 500
180 CLS:LOCATE 10,22:PRINT"Insérer l'Eprom Source dans son Support.":GOSUB 500
190 CLS:LOCATE 10,24:PRINT"Chargement des Données en Cours..."
200 FOR ADR=0 TO LONGR
210    ADR.H=INT(ADR/256):ADR.B=ADR-(ADR.H*256)
220    OUT PORT.A,ADR.H
230    OUT PORT.C,X+1:OUT PORT.C,X
240    OUT PORT.A,ADR.B
250    POKE ADR,INP(PORT.B)
260 NEXT ADR
270 OUT PORT.C,0:BEEP
280 CLS:LOCATE 10,24:PRINT"Insérer l'Eprom Vierge à Programmer.":GOSUB 500
290 CLS:LOCATE 10,22:PRINT"Sélectionner la Tension de Programation.":GOSUB 500
300 T=(INP(PORT.C) AND 96):IF T=0 THEN TENSION=5
310 IF T=32 THEN TENSION=12.5
320 IF T=64 THEN TENSION=21
330 IF T=96 THEN TENSION=25
340 CLS:LOCATE 10,20:PRINT"La Tension de Programmation est de : ";TENSION"V."
350 LOCATE 12,24:INPUT"Confirmation de la Tension : (O/N)?",SP$
360 IF SP$="o" OR SP$="O" GOTO 370 ELSE 290
370 OUT REG.CON,MODE.ECRITURE:OUT PORT.C,9:FOR I=0 TO 1000:NEXT I
380 CLS:LOCATE 10,18:PRINT"Eprom Destination en Cours de Programmation..."
390 FOR ADR=0 TO LONGR
400    ADR.H=INT(ADR/256):ADR.B=ADR-(ADR.H*256)
410    OUT PORT.A,ADR.H
420    OUT PORT.C,10
430    OUT PORT.A,ADR.B
440    OUT PORT.B,PEEK (ADR)
450    OUT PORT.C,9
460    IF (INP(PORT.C) AND 16)=0 GOTO 460
470 NEXT ADR
```

```
480 OUT REG.CON,MODE.LECTURE
490 CLS:BEEP:LOCATE 12,26:PRINT"Fin de la Programmation.":DEF SEG:END
500 LOCATE 12,27:PRINT"<BARRE ESPACE> Une fois Pret..."
510 SP$=INKEY$:IF SP$<>" " THEN GOTO 510
520 RETURN
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : l'instruction DEF SEG permet de définir une zone mémoire à partir de l'adresse 8000 (hexa.) pour stocker les données de l'EPROM à dupliquer. En mode écriture, la configuration de la carte PIA change. Le Port B est programmé en sortie,

lignes 60 à 150 : chargement des valeurs de configuration X et LONGR en fonction du type d'EPROM,

lignes 160 à 190 : configuration du PIA en mode lecture et chargement du port C avec la valeur X,

lignes 200 à 260 : à chaque passage dans la boucle For ... Next, il y a adressage de l'EPROM et chargement de sa donnée dans la zone mémoire de l'ordinateur définie au préalable par l'instruction DEF SEG. Puis, retour au début de la boucle avec incrémentation de l'adresse de l'EPROM et de la zone mémoire jusqu'à chargement complet des données dans la mémoire de l'ordinateur,

lignes 270 à 360 : ces instructions permettent de sélectionner manuellement et de confirmer logiciellement la tension de programmation (V_{pp}) avant le chargement des données dans l'EPROM vierge,

lignes 370 à 380 : configuration du programmateur d'EPROM en mode écriture : relais alimenté et voyant allumé, puis temporisation nécessaire pour la stabilité des contacts électriques du relais,

lignes 390 à 470 : chargement des données présentes en zone mémoire de l'ordinateur dans l'EPROM grâce à l'instruction PEEK. Puis après le déclenchement du temps de programmation (ligne 450), attente de l'écoulement de cette temporisation PGM avant de charger une autre donnée à une nouvelle adresse dans l'EPROM,

lignes 480 à 520 : passage en mode lecture, le voyant s'éteint et fin de la programmation.

PROGRAMME "ROMVERIF"

Ce programme permet de comparer le contenu de deux EPROMS et d'indiquer à l'utilisateur si elles sont identiques ou pas.

```
10  REM -------------------- "ROMVERIF" --------------------
20  CLS:KEY OFF:DEF SEG=&H8000
30  LOCATE 10,22:PRINT"Ne Pas Mettre d'Eprom dans le Support.":GOSUB 380
40  PORT.A=&H310:PORT.B=&H311:PORT.C=&H312
50  REG.CON=&H313:MODE.LECTURE=&H8A
60  CLS:LOCATE 1,14:PRINT"***    Test de Vérification d'une Eprom.    ***"
70  LOCATE 10,24:PRINT"<1>...Eprom de 64 Ko."
80  LOCATE 11,24:PRINT"<2>...Eprom de 128 Ko."
90  LOCATE 12,24:PRINT"<3>...Eprom de 256 Ko."
100 LOCATE 13,24:PRINT"<4>...Eprom de 512 Ko."
110 LOCATE 17,30:INPUT"Choix : ",TEST
120 IF TEST=1 THEN X=0:LONGR=8191:GOTO 160
130 IF TEST=2 THEN X=0:LONGR=16383:GOTO 160
140 IF TEST=3 THEN X=2:LONGR=32767:GOTO 160
150 IF TEST=4 THEN X=6:LONGR=65535!:GOTO 160 ELSE 110
160 OUT REG.CON,MODE.LECTURE:OUT PORT.C,X
170 CLS:LOCATE 10,12:PRINT"Mettre le Commutateur sur le type d'Eprom selectionné
e.":GOSUB 380
180 CLS:LOCATE 10,22:PRINT"Insérer l'Eprom Source dans son Support.":GOSUB 380
190 CLS:LOCATE 10,24:PRINT"Chargement des Données en Cours..."
200 FOR ADR=0 TO LONGR
210     ADR.H=INT(ADR/256):ADR.B=ADR-(ADR.H*256)
220     OUT PORT.A,ADR.H
230     OUT PORT.C,X+1:OUT PORT.C,X
240     OUT PORT.A,ADR.B
250     POKE ADR,INP(PORT.B)
260 NEXT ADR
270 CLS:BEEP:LOCATE 10,29:PRINT"Insérer L'Eprom à Vérifier.":GOSUB 380
280 CLS:LOCATE 10,24:PRINT"Vérification du Contenu de l'Eprom...."
290 FOR ADR=0 TO LONGR
300     ADR.H=INT(ADR/256):ADR.B=ADR-(ADR.H*256)
310     OUT PORT.A,ADR.H
320     OUT PORT.C,X+1:OUT PORT.C,X
330     OUT PORT.A,ADR.B
340     IF INP(PORT.B)<>PEEK (ADR) THEN GOTO 370
350 NEXT ADR
360 CLS:BEEP:LOCATE 10,24:PRINT"Les Contenus sont Identiques!.":DEF SEG:END
370 CLS:BEEP:LOCATE 10,24 : PRINT"Erreur de Vérification!.":DEF SEG:END
380 LOCATE 12,27:PRINT"<BARRE ESPACE> Une fois Pret..."
390 SP$=INKEY$:IF SP$<>" " THEN GOTO 390
400 RETURN
```

EXPLICATION DU PROGRAMME

lignes 10 à 260 : déroulement du logiciel identique au programme "ROMCOPIE". Chargement du contenu d'une EPROM dans la zone mémoire de l'ordinateur,

lignes 270 à 280 : ces instructions permettent d'insérer l'EPROM à vérifier après avoir retiré l'EPROM source,

lignes 290 à 350 : comparaison de la donnée de l'EPROM à vérifier avec la donnée présente en zone mémoire de l'ordinateur pour une même adresse effective et saut à la ligne 370 dans le cas où les contenus sont différents. Cette comparaison s'effectue à toutes les adresses de l'EPROM sélectionnée, si les contenus sont identiques,

lignes 360 à 400 : affichage de la vérification et fin du programme.

⑲ Barrière et télécommande infrarouge

Ce montage, raccordé à un ordinateur (avec la carte PIA/PC et la carte 4E/4S), permet de compter le nombre de personnes ou d'objets ayant coupé le faisceau infrarouge, ou de détecter un passage.

Le circuit émetteur peut être également monté dans un boîtier avec pile 9 V, permettant ainsi d'utiliser le même montage en télécommande infrarouge.

PRINCIPE

L'émetteur est constitué d'un oscillateur commandant deux Leds infrarouges émettrices. Cette fréquence porteuse (environ 10 kHz) permet d'éviter tout déclenchement intempestif dû à des rayonnements parasites (50 Hz, tubes néons).

Grâce à un ampli sélectif et à un circuit de détection, le récepteur ne détecte que toutes coupures de faisceau, ou toutes émissions d'infrarouge modulé.

PRINCIPE BARRIÈRE INFRAROUGE

L'émetteur situé dans le même boîtier que le récepteur émet en permanence un signal infrarouge allant se réfléchir sur un miroir situé à quelques mètres avant d'être détecté par le phototransistor du récepteur. La coupure du faisceau entraîne une non détection de la porteuse, donc le signal de sorties (S) passe au niveau haut (+ 12 V).

PRINCIPE TÉLÉCOMMANDE INFRAROUGE

Chaque action du bouton poussoir permet d'émettre un signal infrarouge qui active la sortie "S" du récepteur au niveau bas (0 V). Après avoir désactivé le bouton poussoir, l'émission infrarouge est coupée et le signal "S" repasse au niveau haut (+ 12 V).

TECHNOLOGIE

Récepteur

Une ajustable permet de régler le courant de polarisation du phototransistor en fonction de la luminosité ambiante.

Le signal issu du phototransistor est amplifié et filtré par deux amplis opérationnels (741) avec une fréquence de coupure identique pour les filtres passe haut et passe bas (environ 10 kHz). Le signal obtenu est redressé par deux diodes (1N 4148) et un condensateur avant de commander la base d'un transistor NPN.

La détection d'un faisceau infrarouge modulé entraîne la charge du condensateur de 0,47 µF, puis la saturation du transistor NPN (signal "S" niveau bas).

Émetteur

Un circuit intégré NE 555 permet d'obtenir une fréquence stable (10 kHz) quelle que soit la tension d'alimentation (pile 9 V ou + 12 V) de la carte 4 Entrées/4 Sorties. Ce signal disponible patte 3 du CI est amplifié par un transistor PNP avant de commander les Leds émettrices infrarouges. La résistance R_1 permet de limiter le courant dans les Leds infrarouges et de recharger le condensateur C_1 après relâchement du BP.

UTILISATION

Barrière infrarouge (*fig.* 19.1)

Remarque : L'entrée E_1 (PB0) de la carte 4E/4S doit être raccordée à la sortie "S" du récepteur.

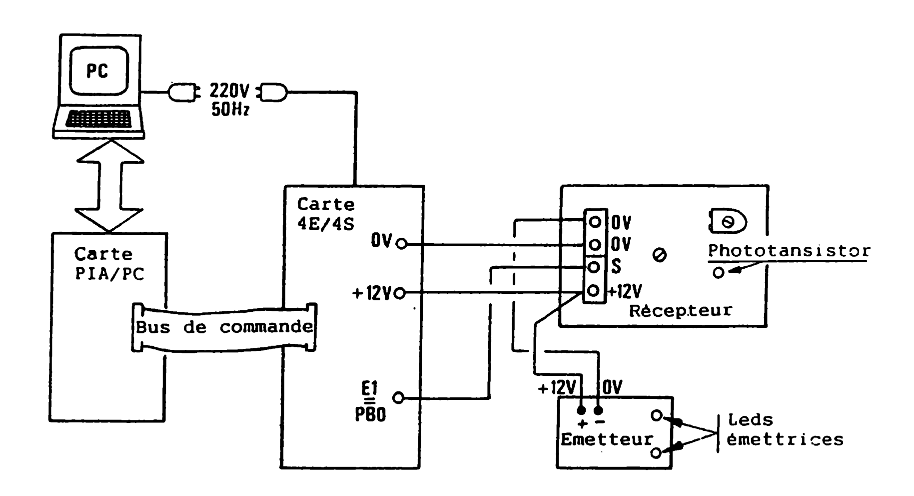


Fig. 19.1

Télécommande infrarouge (*fig.* 19.2)

Remarque : Pour cette utilisation, souder le condensateur de 100 µF (C_1) en respectant la polarité, et changer la résistance R_1 par une résistance de valeur 10 Ω.

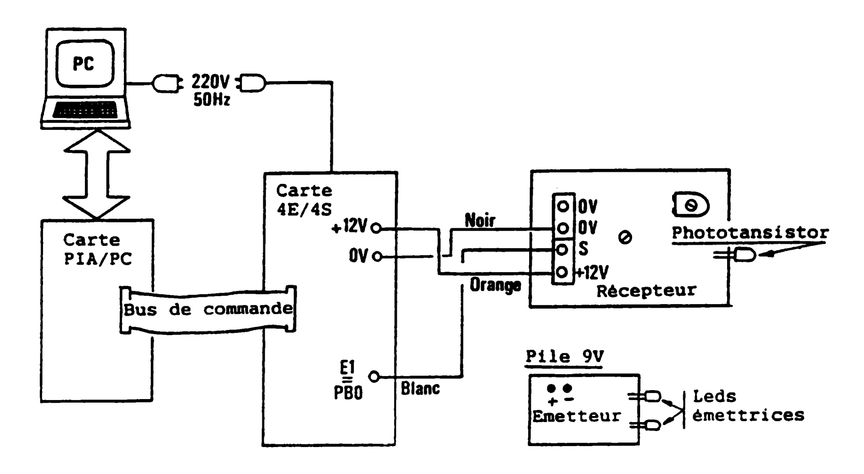
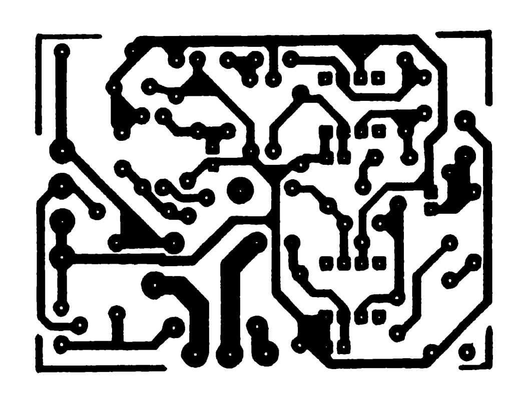


Fig. 19.2

CIRCUITS CÔTÉ CUIVRE



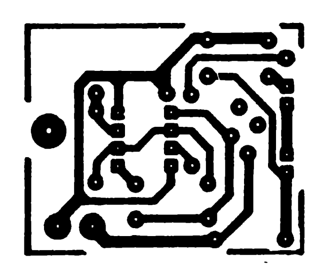


Fig. 19.3

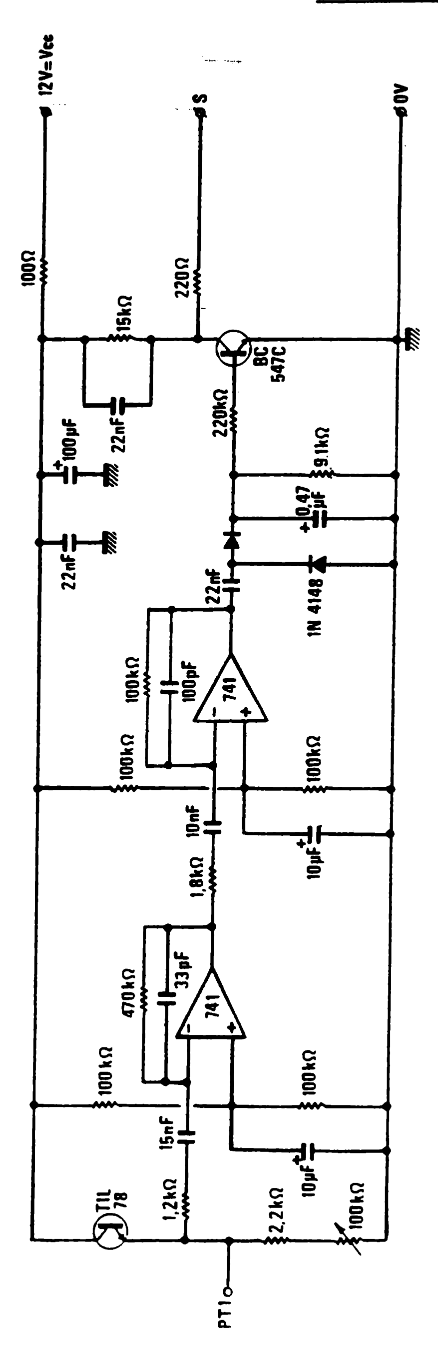

12V=Vcc
S
0V
100Ω
220Ω
15kΩ
BC 547C
100µF
22nF
22nF
220kΩ
9,1kΩ
0,47 µF
22nF
1N 4148
100kΩ
100pF
741
100kΩ
100kΩ
10nF
10µF
1,8kΩ
470kΩ
33pF
741
100kΩ
100kΩ
15nF
10µF
TIL 78
1,2kΩ
2,2kΩ
100kΩ
PT1

Fig. 19.4a

SCHÉMAS DE PRINCIPE (*fig.* 19.4)

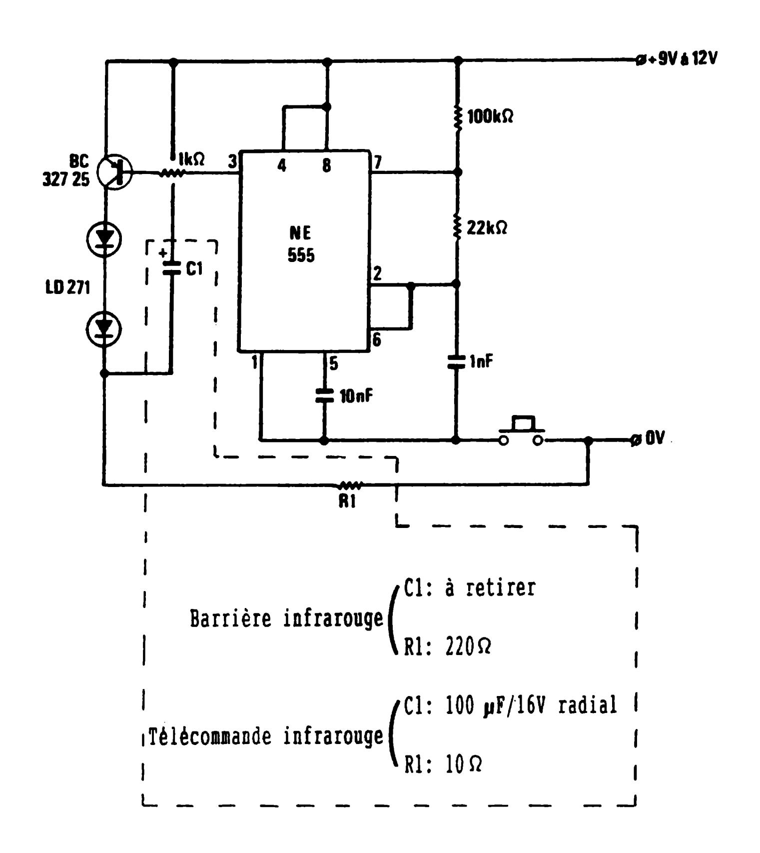


Fig. 19.4b

MONTAGE

Le récepteur (*fig.* 19.5)

Souder dans l'ordre suivant et le plus près possible du circuit :

— les deux diodes dont le sens est repéré par une bague,

— les résistances,

— les condensateurs (attention à la polarité des chimiques),

— la résistance ajustable,

— le transistor dont le sens est repéré par un méplat,

— les 741 repérés par un point ou une encoche,

— le TIL 78 (attention patte longue au point E),

— les deux borniers à vis (sorties fils vers l'extérieur du circuit),

— à l'aide de fils isolés, relier le point + 12 V de la carte 4E/4S au + 12 V du récepteur, le point 0 V au point 0 V et le point E_1 au S.

Récepteur

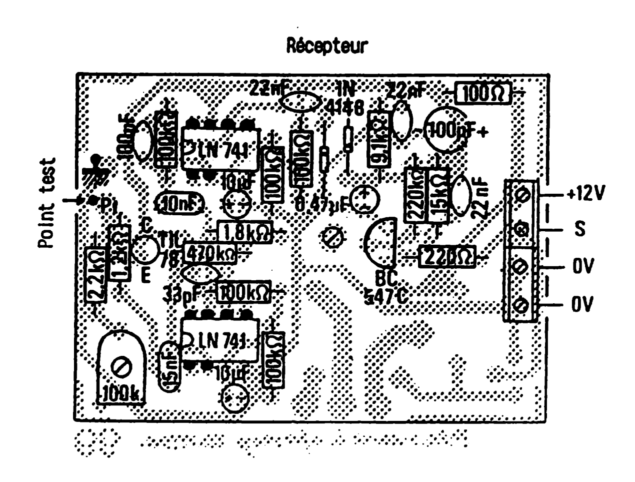


Fig. 19.5

L'émetteur télécommande (*fig.* 19.6)

— les résistances,

— les condensateurs (attention à la polarité du chimique),

— le transistor dont le sens est repéré par un méplat,

— le NE 555 repéré par un point ou une encoche,

— les deux LED patte longue au plus,

— le bouton-poussoir,

— câbler les fils du clips de pile (attention à la couleur : voir implantation).

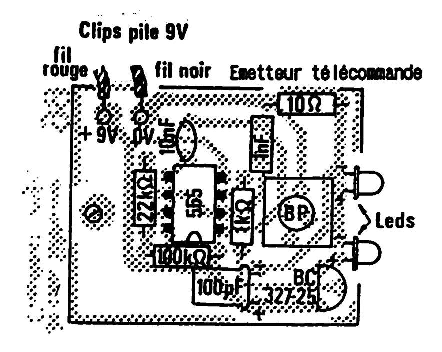


Fig. 19.6

L'émetteur barrière infrarouge (*fig.* 19.7)

— le strap,

— les résistances,

— les condensateurs qui seront implantés à plat,

— le transistor dont le sens est repéré par un méplat qui lui aussi devra être couché sur le circuit,

— le NE 555 repéré par un point ou une encoche,

— les deux LED patte longue au plus,

— souder les deux fils isolés de longueur 10 cm sur les points + 12 V et 0 V que vous câblerez sur le bornier à vis du récepteur sur les points + 12 V et 0 V.

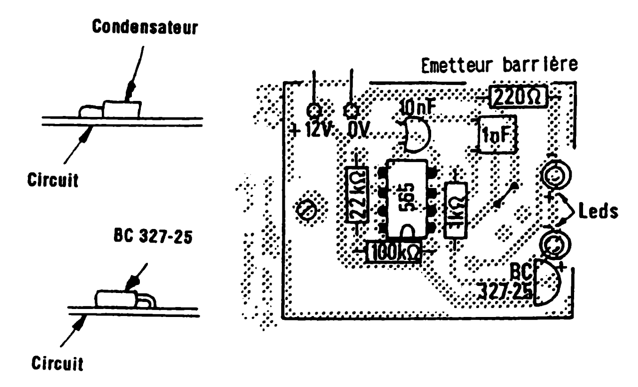


Fig. 19.7

Remarque : Dans le cas d'un montage en barrière infrarouge, ne pas monter le condensateur de 100 µF (C_1), remplacer les contacts du BP par un strap (queue de résistance); la valeur de R_1 est de 220 Ω.

Coucher les condensateurs et le transistor sur le circuit imprimé.

RÉGLAGE (*fig.* 19.8)

Placer le récepteur sans mettre la face avant du boîtier à l'endroit désiré.

Alimenter uniquement le récepteur par l'intermédiaire de l'alim. de la carte 4E/4S et régler à l'aide d'un voltmètre l'ajustable pour obtenir une tension continue d'environ 6 V ($V_{CC}/2$) entre masse et point test (Pt).

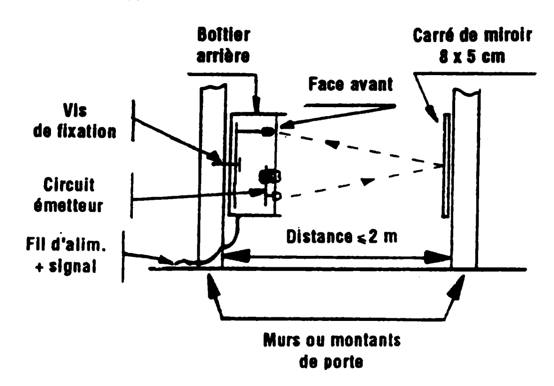


Fig. 19.8

MISE EN BOÎTIER DE LA BARRIÈRE (*fig.* 19.9)

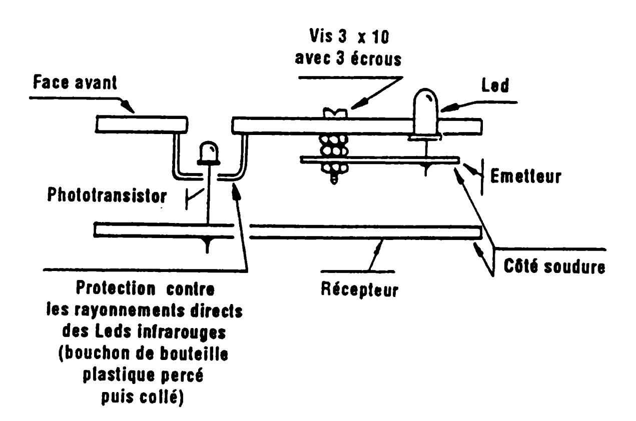


Fig. 19.9

MISE EN BOÎTIER DE L'ÉMETTEUR TÉLÉCOMMANDE (*fig.* 19.10)

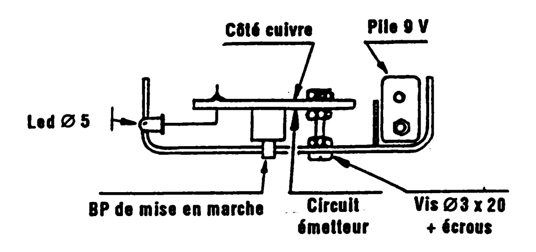


Fig. 19.10

LISTE DES COMPOSANTS DU RÉCEPTEUR

Résistances

1 × 100 Ω (marron noir marron),
1 × 220 Ω (rouge rouge marron),
1 × 1,2 kΩ (marron rouge rouge),
1 × 1,8 kΩ (marron gris rouge),
1 × 2,2 kΩ (rouge rouge rouge),
1 × 9,1 kΩ (blanc marron rouge),
1 × 15 kΩ (marron vert orange),
5 × 100 kΩ (marron noir jaune),
1 × 220 kΩ (rouge rouge jaune),
1 × 470 kΩ (jaune violet jaune).

Condensateurs

1 × 33 pF,
1 × 100 pF,
1 × 10 nF MKT,
1 × 15 nF MKT,
3 × 22 nF céramique,
1 × 0,47 µF chimique, radial,
2 × 10 µF/16 V radial,
1 × 100 µF/16 V radial.

Semiconducteurs

1 × TIL 78 phototransistor,
2 × 741,
2 × 1N 4148,
1 × BC 547 C.

Divers

2 × bornier à vis pour CI,
1 × ajustable 100 k horizontal,
1 × vis 3 × 10 + écrou,
1 × boîtier 10P avec face avant et vis.

LISTE DES COMPOSANTS DE L'ÉMETTEUR TÉLÉCOMMANDE

Résistances

1 × 10 Ω (marron noir noir),
1 × 1 kΩ (marron noir rouge),
1 × 22 kΩ (rouge rouge orange),
1 × 100 kΩ (marron noir jaune),
1 × NE 555,
1 × BC 327-25,
2 × LD 271,
1 LED infrarouge,
1 × 1 nF MKT,
1 × 10 nF céramique,
1 × 100 µF/16 V axial,
1 × bouton poussoir pour CI,
1 × clips de pile 9 V,
1 × boîtier DBC,
1 × vis 3 × 20 et trois écrous.

LISTE DES COMPOSANTS DE L'ÉMETTEUR BARRIÈRE INFRAROUGE

1 × 220 Ω (rouge rouge marron),
1 × 1 kΩ (marron noir rouge),
1 × 22 kΩ (rouge rouge orange),
1 × 100 kΩ (marron noir jaune),
1 × NE 555,
1 × BC 327-25,
1 LED infrarouge,
2 × LD 271,
1 × 1 nF MKT,
1 × 10 nF céramique,
1 × vis 3 × 10 et trois écrous.

PROGRAMME "INFRABAR"

PROGRAMME

Ce programme écrit en GW BASIC de Microsoft permet de compter le nombre de personnes ou d'objets et de créer un bip sonore à chaque coupure du faisceau.

Ce logiciel permet également, grâce à l'action de la touche "A" du clavier de l'ordinateur, de désactiver le bip sonore ou de remettre à zéro le comptage touche "R".

```
5   REM ---------- programme ---------- infrabar --------
10  CLS:KEY OFF:NBR=0:B=1
20  LOCATE 3,15:PRINT "********** Comptage barriere infrarouge **********"
30  LOCATE 19,24:PRINT "(R) : Remise à zèro du comptage"
40  LOCATE 21,29:PRINT "(F) : Fin du comptage"
50  LOCATE 23,30:PRINT "(M/A) : Beep sonore"
60  PORT.A=&H310:PORT.B=&H311
70  REG.CONTROLE=&H313:OUT REG.CONTROLE,&H8B
80  LOCATE 10,10:PRINT "Nombre de personne (ou d'objet) ayant coupé le faisceau
:";NBR
90  IF (INP(PORT.B) AND 1)=1 GOTO 90
100 FOR X=1 TO 1500:NEXT X
110 IF (INP(PORT.B) AND 1)=1 GOTO 180
120 A$=INKEY$
130 IF A$="r" OR A$="R" THEN NBR=0:GOTO 80
140 IF A$="f" OR A$="F" THEN CLS:END
150 IF A$="m" OR A$="M" THEN B=1
160 IF A$="a" OR A$="A" THEN B=0
170 GOTO 110
180 IF B=1 THEN BEEP
190 NBR=NBR+1
200 GOTO 80
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : effacement et affichage de message à l'écran. La variable B = 1 permet d'activer le bip sonore dès le lancement du programme. Si à la ligne 10, B = 0, il n'y a pas d'activation du bip sonore. L'action de la touche "M" permet de réactiver le bip sonore une fois le programme lancé.

lignes 60 à 80 : initialisation de la carte PIA/PC, Port A en sortie et Port B en entrée; Affichage à l'écran du nombre de personnes ayant coupé le faisceau.

lignes 90 à 100 : tant que le faisceau est coupé (PB0 = 1), le programme reste à la ligne 90. Cette détection de front descendant permet d'attendre que la personne (ou l'objet) ait totalement traversé le faisceau, donc de ne pas compter plusieurs fois la même personne. La temporisation ligne 100 est nécessaire

afin d'éviter tout déclenchement intempestif dû à la forme de l'objet ou à la vitesse de passage d'une personne devant le faisceau infrarouge.

lignes 110 à 170 : dès la coupure du faisceau infrarouge (PB0 = 1), le programme va à la ligne 180. Sans coupure de faisceau, le programme analyse si une touche du clavier est actionnée et revient à la ligne 110.

lignes 180 à 200 : l'action de la touche "M" ou "A" charge la variable B avec la valeur 1 ou 0, permettant de faire retentir ou pas le bip sonore. Après incrémentation de la variable NBR, retour en début de programme ligne 80.

PROGRAMME "INFRATEL"

PROGRAMME

Ce logiciel simple permet d'alimenter ou de couper l'alimentation du Relais n.1 de la carte 4E/4S à chaque impulsion du bouton-poussoir de l'émetteur infrarouge. Lorsque le récepteur détecte une impulsion d'émission, l'ordinateur fait retentir un bip sonore. Ce bip peut être désactivé par l'action de la touche "A" du clavier de l'ordinateur, ou réactivé par la touche "M".

```
5   REM ---------- programme ---------- infratel ----------
10  CLS:KEY OFF:B=1:X=0
20  LOCATE 3,18:PRINT "********** Télécommande infrarouge **********"
30  LOCATE 20,33:PRINT "(Q) : Quitter"
40  LOCATE 22,31:PRINT "(M/A) : Beep sonore"
50  REG.CONTROLE=&H313
60  PORT.A=&H310:PORT.B=&H311
70  OUT REG.CONTROLE,&H8B
80  IF X=1 THEN LOCATE 12,30:PRINT " Relais N°:1 activé ! " ELSE LOCATE 12,30:PR
INT "Relais N°:1 au repos !"
90  IF (INP(PORT.B) AND 1)=0  GOTO 90
100 FOR T=0 TO 1000:NEXT T
110 IF (INP(PORT.B) AND 1)=0  GOTO 170
120 A$=INKEY$
130 IF A$="m" OR A$="M" THEN B=1
140 IF A$="a" OR A$="A" THEN B=0
150 IF A$="q" OR A$="Q" THEN CLS:END
160 GOTO 110
170 IF B=1 THEN BEEP
180 IF X=1 THEN X=0 ELSE X=1
190 OUT PORT.A,X
200 GOTO 80
```

EXPLICATION DU PROGRAMME

lignes 10 à 80 : effacement et affichage de message à l'écran. Chargement de la variable B avec la valeur 1 permettant d'activer le bip sonore au lancement du programme et chargement de X avec la valeur zéro permettant de laisser au repos le relais de commande. Initialisation de la carte PIA/PC, Port A en sortie et Port B en entrée. Puis affichage à l'écran de l'état du relais,

lignes 90 à 170 : lignes identiques aux lignes 90 à 170 du programme INFRABAR,

ligne 180 : ces instructions permettent de changer la valeur de la variable X par 0 ou par 1 à chaque fois que l'action du bouton poussoir de l'émetteur est détecté,

lignes 190 à 200 : chargement du Port A avec la valeur de la variable X et retour au début du programme.

Remarque : Pour activer un autre relais, changer la valeur X = 1 par X = 4 pour le relais numéro 3 par exemple.

⑳ Télécommande téléphonique

PRINCIPE (*fig.* 20.1)

Le montage permet la mise en route à distance de votre ordinateur grâce au téléphone.

Il suffit de téléphoner chez soi trois à quatre fois de rang à 30 secondes d'intervalle en laissant sonner deux fois à chaque appel.

Ce montage permet également, par l'intermédiaire de la carte PIA/PC et de la carte 4 Entrées/4 Sorties, de couper l'ordinateur après la fin du traitement d'un programme (mise en route du chauffage, simulation de présence, arrosage automatique...).

Appel du programme dans AUTOEXECT.BAT).

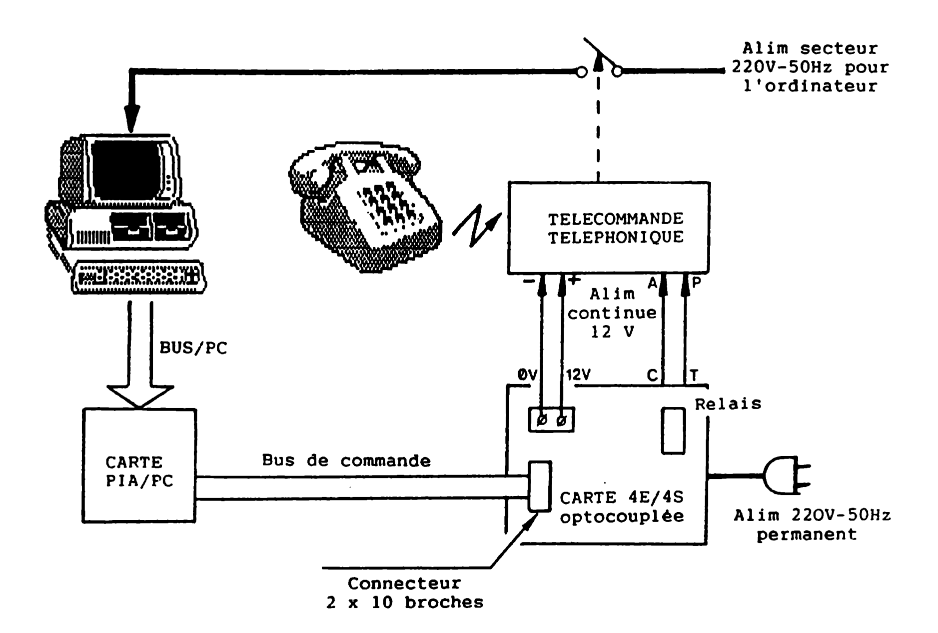


Fig. 20.1

SCHÉMA DE PRINCIPE (*fig.* 20.2)

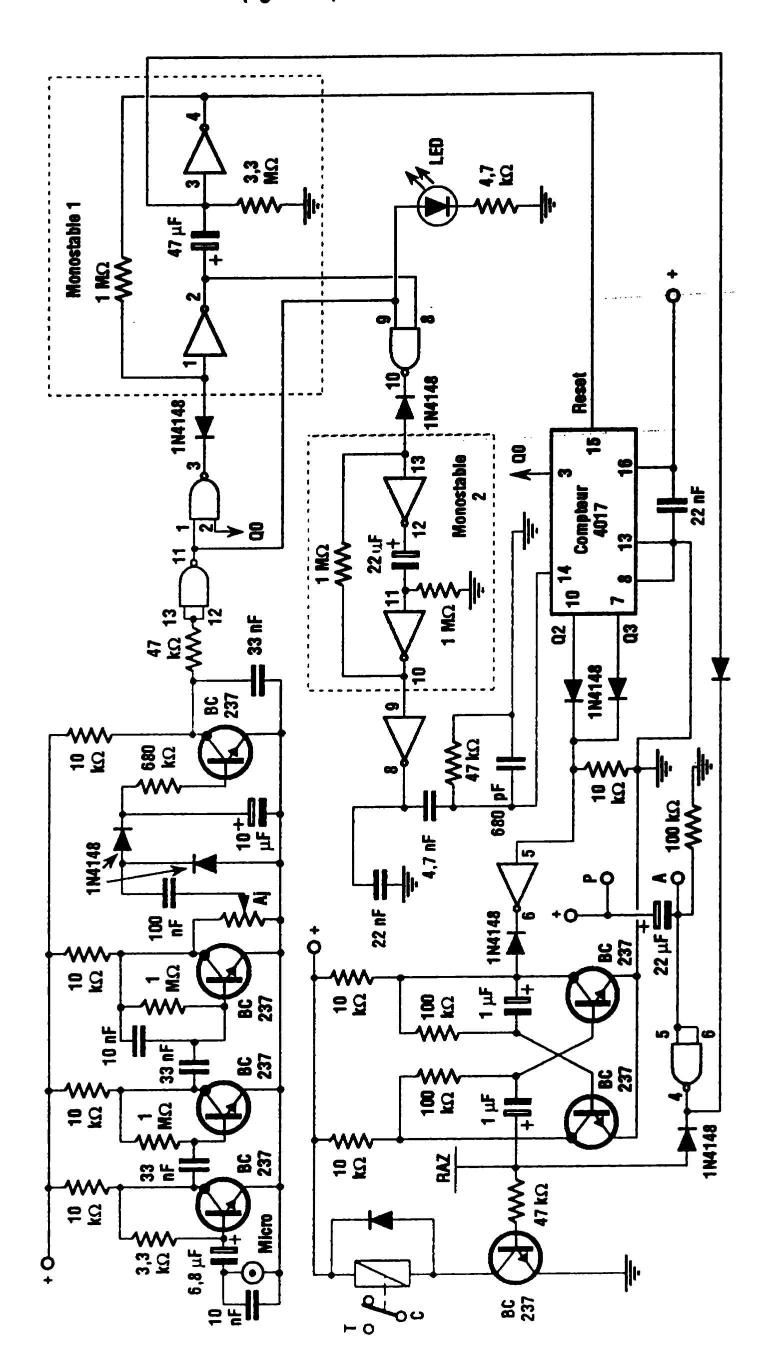


Fig. 20.2

CIRCUIT IMPRIMÉ CÔTÉ CUIVRE

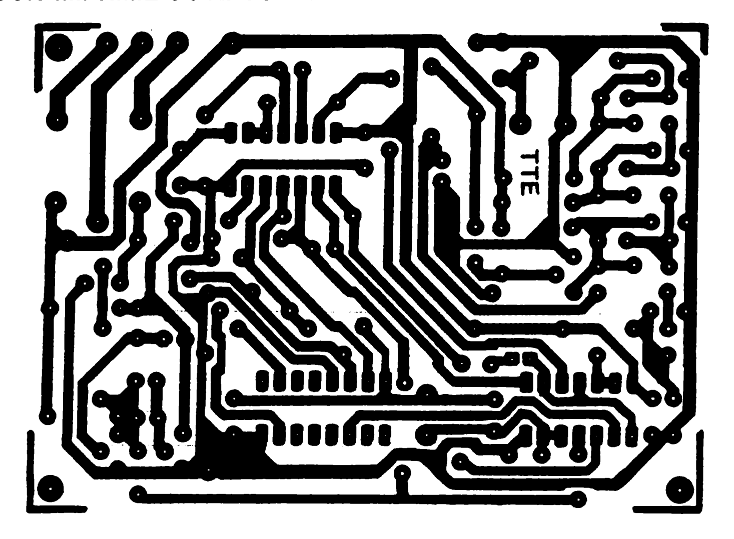


Fig. 20.3

IMPLANTATION

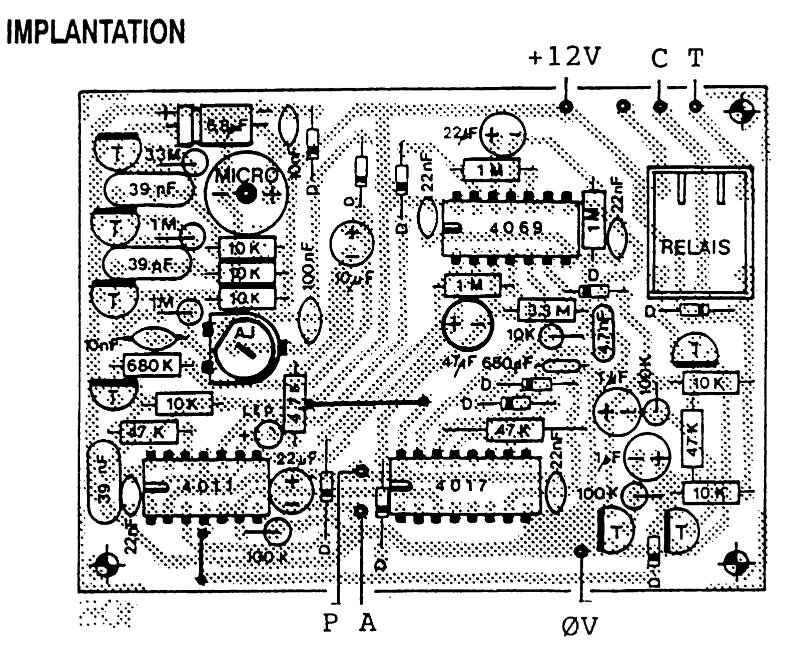
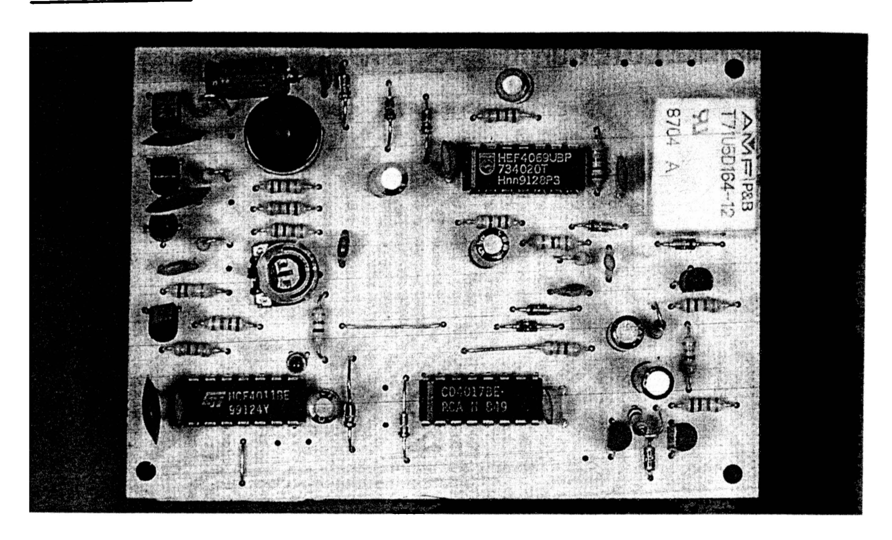


Fig. 20.4



LISTE DES COMPOSANTS

Résistances

1 × 4,7 kΩ (jaune violet rouge),
7 × 10 kΩ (marron noir orange),
3 × 47 kΩ (jaune violet orange),
3 × 100 kΩ (marron noir jaune),
1 × 680 kΩ (bleu gris jaune),
5 × 1 MΩ (marron noir vert),
2 × 3,3 MΩ (orange orange vert).

Semiconducteurs

10 × 1N 4148,
7 × BC 548 C ou 237 C,
1 × CD 4069,
1 × CD 4011,
1 × CD 4017.

Condensateurs

1 × 680 pF,
1 × 4,7 nF,
2 × 10 nF,
4 × 22 nF,
3 × 47 nF,
1 × 0,1 µF,
2 × 1 µF/50 V radial,
1 × 6,8 µF axial ou 4,7 µF,
1 × 10 µF/25 V radial,
1 × 22 µF/10 V radial,
1 × 47 µF/16 V radial.

Divers

un relais 12 V/3 A (AMF),
une LED rouge diamètre 3,
un micro,
une ajustable horizontal 220 kΩ.

MONTAGE

Souder dans l'ordre et le plus près possible du circuit :

— les deux straps repérés •—•,

— les résistances; certaines sont implantées verticalement et repérées —o sur l'implantation,

— les diodes dont le sens est indiqué par la bague,

— l'ajustable,

— les condensateurs en respectant la polarité des chimiques,

— les sept transistors,

— la LED : attention à son sens, patte longue au plus,

— les circuits intégrés repérés par un point ou une encoche,

— le micro en respectant la polarité,

— le relais.

BRANCHEMENT

Connecter la carte 4 entrées/4 sorties à votre ordinateur (voir notice carte 4 entrées/4 sorties).

Relier les points marqués 0 V et + 12 V au bornier de la carte 4 entrées/ 4 sorties repéré 0 V et + 12 V en respectant la polarité. Relier le point A au point C_1 de la carte 4 entrées/4 sorties et le point P au point P_1. Prévoir des longueurs de fils suffisantes pour pouvoir éloigner les deux cartes d'au moins 50 cm pour éviter que le micro ne capte le rayonnement 50 Hz du transformateur (voir figure 20.1).

Câbler l'alimentation secteur de votre ordinateur comme indiqué sur la figure 20.5.

L'interrupteur de mise en marche de l'ordinateur doit être bien entendu basculé sur la position marche.

C'est le relais de la télécommande qui lui servira pour la mise en route.

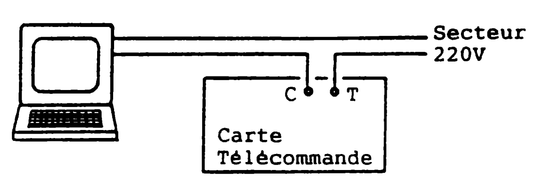


Fig. 20.5

FONCTIONNEMENT

Le son de la sonnerie est capté par un micro amplifié et détecté. Lors de la première sonnerie, on déclenche un monostable (M_1) de durée supérieure à 2 min. La sonnerie et ce monostable déclenchent un deuxième monostable (M_2) de durée environ 30 s qui incrémente un compteur (4017).

Si pendant ce temps de 30 s une autre sonnerie intervient, elle est sans effet sur M_2.

Les 30 s écoulées, une nouvelle sonnerie déclenchera M_2 et une nouvelle incrémentation. Après trois cycles, la sortie Q_2 du compteur validera une bascule (Flip-Flop) qui fera coller le relais, alimentant ainsi le PC.

Une sonnerie permanente maintient M_2 déclenché et empêche ainsi l'incrémentation. Si pendant 2 min il n'y a plus de sonnerie, le montage remet le compteur à zéro.

Remarque : Pour plus de sécurité, la sortie Q_3 du compteur (4017) est également détectée. Cela vous permettra de faire quatre appels espacés de 30 s afin d'être certain que le PC est alimenté (*fig.* 20.6).

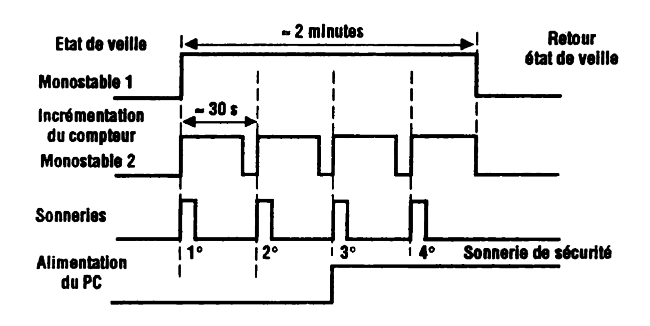


Fig. 20.6

FONCTIONNEMENT DE L'AUTOCOUPURE

Suite à une mise en route de l'ordinateur par téléphone, et si on relie les points repérés P et A (voir figure 20.1), cela aura pour effet de désactiver le relais. Cette commande peut être réalisée par les contacts commun (C) et travail (T) d'un des 4 relais de la carte 4 Entrées/4 Sorties. Elle vous permettra après avoir exécuté un programme d'arrosage automatique par exemple de couper l'alimentation de votre ordinateur par le logiciel assurant ainsi une économie d'énergie (*fig.* 20.7).

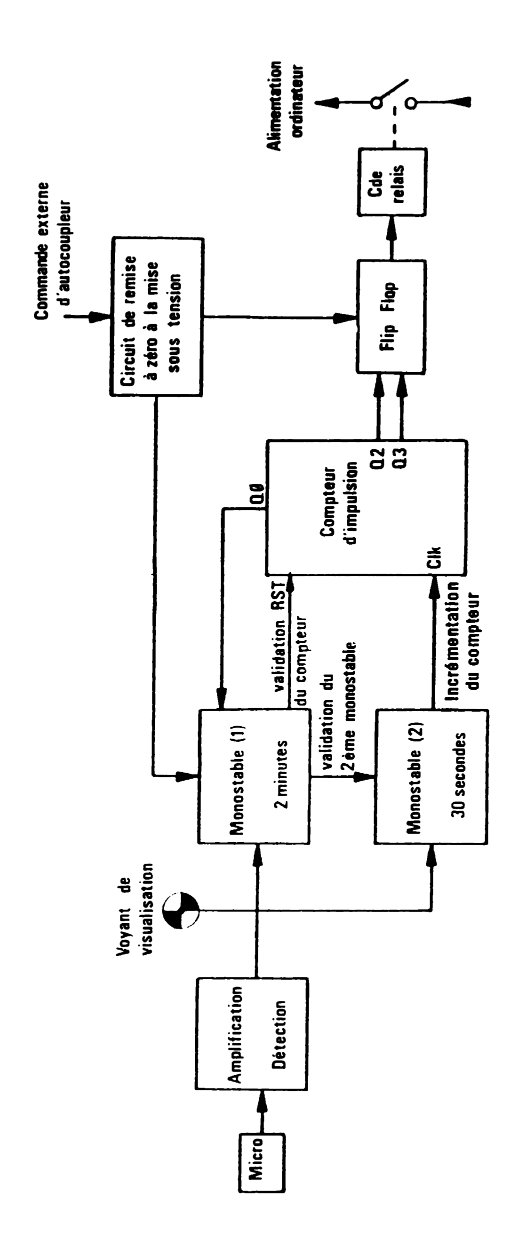

Micro
Amplification
Détection
Voyant de visualisation
Monostable (1)
2 minutes
Monostable (2)
30 secondes
validation du 2ème monostable
validation du compteur
RST
Incrémentation du compteur
Clk
Compteur d'impulsion
Q0
Q2
Q3
Circuit de remise à zéro à la mise sous tension
Commande externe d'autocoupleur
Flip Flop
Cde relais
Alimentation ordinateur

Fig. 20.7

PROGRAMMATION

Dès la mise sous tension et après boutage, l'ordinateur exécute automatiquement le contenu du fichier : "AUTOEXEC.BAT".

Pour lancer automatiquement un logiciel (arrosage...), il suffit de rajouter le nom de ce logiciel à la fin du fichier "AUTOEXEC.BAT".

INSTRUCTIONS UTILES

TYPE AUTOEXEC.BAT puis touche entrée (enter) permet de lister le contenu de ce fichier sur l'écran.

EDLIN AUTOEXEC.BAT puis touche entrée (enter) permet de modifier le contenu de ce fichier (confér doc. du DOS).

Pour l'autocoupure de votre ordinateur, il suffit d'activer le relais (contact C --- A et T --- P) relié à la carte télécommande téléphonique (voir documentation carte 4 Entrées/4 Sorties).

CARACTÉRISTIQUES TECHNIQUES

un contact relais 1,5 A/220 V,

une tension d'alimentation 12 V,

une consommation inférieure à 100 mA,

une LED de visualisation de la sonnerie.

RÉGLAGE

Alimenter la carte "télécommande téléphonique" par le 12 V de la carte 4 Entrées/4 Sorties en respectant les polarités.

Mettre la carte "télécommande téléphonique" près de la sonnerie du téléphone, éloigner d'environ 50 cm la carte 4 Entrées/4 Sorties (rayonnement 50 Hz).

Tourner l'ajustable de telle sorte qu'à chaque sonnerie de téléphone, la LED s'allume et s'éteint quand la sonnerie s'arrête.

㉑ Annexes

1. PROGRAMME "ENTREE"

Avec la carte 4 entrées/4 sorties.

TEST DES ENTRÉES

Sous GW BASIC, faire RUN"ENTREE".

Le programme teste les quatre entrées de la carte 4 entrées/4 sorties et indique si celles-ci sont au repos ou activées.

Par exemple, en réunissant les deux plots de l'entrée 4, le programme nous indiquera *entrée 4 activée.* Si l'on met une photorésistance entre les deux points du bornier 4, en éclairant celle-ci avec une lampe et en réglant l'ajustable 4, à un certain seuil l'ordinateur indiquera *entrée 4 activée.* On peut donc détecter une certaine luminosité réglable grâce à l'ajustable (ou autre : température, etc.).

```
10 '---programme---entree---
20 CLS:KEY OFF
30 OUT &H313,&H82:PORTB=&H311:PORTA=&H310
40 COLOR 7,0:LOCATE 6,35:PRINT "Test des entrées "
50 LOCATE 25,20:PRINT" Retour au BASIC appuyer sur barre espace."
60 U=INP(PORTB)
70 LOCATE 12,2 :PRINT "entrée 1:";
80 IF(U AND 1)=1 THEN PRINT " activée ";:GOTO 100
90 COLOR 0,7:PRINT " au repos";:COLOR 7,0
100 LOCATE 14,2 :PRINT "entrée 2:";
110 IF(U AND 2)=2 THEN PRINT " activée ";:GOTO 130
120 COLOR 0,7:PRINT " au repos";:COLOR 7,0
130 LOCATE 16,2 :PRINT "entrée 3:";
140 IF(U AND 4)=4 THEN PRINT " activée ";:GOTO 160
150 COLOR 0,7:PRINT " au repos";:COLOR 7,0
160 LOCATE 18,2 :PRINT "entrée 4:";
170 IF(U AND 8)=8 THEN PRINT " activée ";:GOTO 190
180 COLOR 0,7:PRINT " au repos";:COLOR 7,0
190 A$=INKEY$
200 IF A$=" " THEN COLOR 7,0:CLS:OUT PORTA,0:END
210 GOTO 60
```

EXPLICATION DU PROGRAMME

ligne 30 : configuration du PIA, port B en entrée, port A en sortie,

lignes 40 et 50 : affichage texte,

ligne 60 : entrée du port B,

lignes 70 à 90 : test et affichage entrée 1,

lignes 100 à 120 : test et affichage entrée 2,

lignes 130 à 150 : test et affichage entrée 3,

lignes 160 à 180 : test et affichage entrée 4,

lignes 190 et 200 : test si barre espace appuyée; si oui fin du programme,

ligne 210 : retour à entrée port B.

Test des huit entrées/sorties : voir programme ENTSORT.

2. PROGRAMME ARROSÉ

Ce programme permet d'arroser jusqu'à quatre zones différentes commandées par les relais des sorties de la carte 4 entrées/4 sorties : zone 1 = sortie 1, zone 2 = sortie 2, zone 3 = sortie 3, zone 4 = sortie 4. Les entrées 1 à 4 reçoivent un détecteur d'humidité (deux morceaux en inox plantés dans la terre de chaque zone et reliés par des fils aux entrées 1 à 4). Régler le seuil de détection avec les ajustables de la carte 4E/4S. Si on veut arroser quatre zones différentes, lorsque le capteur 1 détectera une humidité suffisante, il validera la zone 2 et ainsi de suite jusqu'à la zone 4.

Si les quatre capteurs détectent une humidité suffisante, l'arrosage s'arrête.

De plus on composera l'heure à laquelle l'arrosage peut commencer et l'heure à laquelle il doit finir. Deux compositions sont possibles : par exemple une pour le soir et une pour le matin.

Composer RUN"ARROSE" sous GW BASIC

Nombre de zones à arroser? Composer par exemple 4 puis entrée (enter).

Heure début cycle 1? Par ex. 193000 (soit 19 h 30 min 00)

Fin de cycle 1? Par ex. 195500 (soit 19 h 55 min 00)

Heure début cycle 2? Composer 000000

Fin de cycle 2 Composer 000000

Soit pas de cycle 2.

L'heure clignote en bas à gauche. À 19 h 30, si l'entrée 1 n'est pas détectée (terre sèche), la sortie 1 sera commandée.

Si l'entrée 1 est détectée (terre suffisamment humide), on passe directement à l'entrée 2 et ainsi de suite jusqu'à 4.

À 19 h 55, l'arrosage sera arrêté quel que soit l'état des capteurs.

Retour au BASIC en appuyant sur barre d'espacement.

```
10 'config pia------------
20 P=&H313
30 O=&H310
40 R=&H311
50 OUT P,&H82
60 COLOR 7,0:CLS
70 KEY OFF
75 '--entrée--constantes--
80 LOCATE 8,15:COLOR 0,7:PRINT " nombre de zones à arroser        ";
90 LOCATE,45:INPUT CY:IF CY>4 GOTO 80
100 LOCATE 10,5:COLOR 0,7:PRINT " Heure début ler cycle   hh:mm:ss ";
110 LONG=2:LIGNE=10:POSITION=30:MINI=0:MAXI=23
120 GOSUB 790
130 A=VAL(BZ$)
140 LONG=2:LIGNE=10:POSITION=33:MINI=0:MAXI=59
150 GOSUB 790
160 AA=VAL(BZ$):A=A*100:A=AA+A
170 LONG=2:LIGNE=10:POSITION=36:MINI=0:MAXI=59
180 GOSUB 790
190 AA=VAL(BZ$):A=A*100:A=AA+A
200 LOCATE 10,40:COLOR 0,7:PRINT " Heure fin ler cycle   hh:mm:ss ";
210 LONG=2:LIGNE=10:POSITION=63:MINI=0:MAXI=23
220 GOSUB 790
230 A1=VAL(BZ$)
240 LONG=2:LIGNE=10:POSITION=66:MINI=0:MAXI=59
250 GOSUB 790
260 AB=VAL(BZ$):A1=A1*100:A1=AB+A1
270 LONG=2:LIGNE=10:POSITION=69:MINI=0:MAXI=59
280 GOSUB 790
290 AB=VAL(BZ$):A1=A1*100:A1=AB+A1
310 LOCATE 12,5:COLOR 0,7:PRINT " Heure début 2ème cycle  hh:mm:ss ";
320 LONG=2:LIGNE=12:POSITION=30:MINI=0:MAXI=23
330 GOSUB 790
340 B=VAL(BZ$)
350 LONG=2:LIGNE=12:POSITION=33:MINI=0:MAXI=59
360 GOSUB 790
370 BB=VAL(BZ$):B=B*100:B=BB+B
380 LONG=2:LIGNE=12:POSITION=36:MINI=0:MAXI=59
390 GOSUB 790
400 BB=VAL(BZ$):B=B*100:B=BB+B
410 LOCATE 12,40:COLOR 0,7:PRINT " Heure fin 2ème cycle  hh:mm:ss ";
420 LONG=2:LIGNE=12:POSITION=63:MINI=0:MAXI=23
430 GOSUB 790
440 B1=VAL(BZ$)
450 LONG=2:LIGNE=12:POSITION=66:MINI=0:MAXI=59
460 GOSUB 790
470 CB=VAL(BZ$):B1=B1*100:B1=CB+B1
480 LONG=2:LIGNE=12:POSITION=69:MINI=0:MAXI=59
490 GOSUB 790
500 CB=VAL(BZ$):B1=B1*100:B1=CB+B1
510 LOCATE 1,15:PRINT" Appuyer sur barre espace pour arrêter le programme "
520 COLOR 7,0
525 '---commande--arrosage---
530 LOCATE 25,2:COLOR 16,7:PRINT TIME$;
540 B$=INKEY$
550 IF B$=" " THEN OUT O,0:COLOR 7,0:CLS:END
555 '--calcul--del'heure---dans--c--
560 C$=LEFT$(TIME$,2)+MID$(TIME$,4,2)+RIGHT$(TIME$,2):C=VAL(C$)
565 '--test--si--heure--dans--constantes--
570 IF C>=A AND C<A1 THEN GOTO 610
580 IF C>=B AND C<B1 THEN GOTO 610
590 OUT O,0
600 GOTO 520
610 COLOR 7,0
615 '--test--des--capteurs--d'entrée----
620 ENT=INP(R)
630 IF (ENT AND 1)=1 THEN E=E AND 14 ELSE GOTO 680
640 OUT O,E:IF (ENT AND 2)=2 THEN E=(E AND 13) ELSE GOTO 700
```

```
650 OUT O,E:IF (ENT AND 4)=4 THEN E=(E AND 11) ELSE GOTO 730
660 OUT O,E:IF (ENT AND 8)=8 THEN E=(E AND 7) :OUT O,E:ELSE GOTO 760
670 GOTO 530
680 E=1:OUT O,E:FOR I=1 TO 500:NEXT I
690 GOTO 530
700 IF CY<2 THEN GOTO 530
710 E=2:OUT O,E:FOR I=1 TO 500:NEXT I
720 GOTO 530
730 IF CY<3 THEN GOTO 530
740 E=4:OUT O,E:FOR I=1 TO 500:NEXT I
750 GOTO 530
760 IF CY<4 THEN GOTO 530
770 E=8:OUT O,E:FOR I=1 TO 500:NEXT I
780 GOTO 530
790 '---entrée -par -inkeys--long=longueur-enre--mini=valeur-mini
800 '-maxi=valeur-maxi--ligne=position-pour-locate-résultat--dans--bz$
810  BZ$=""
820  LOCATE LIGNE,POSITION:PRINT STRING$(LONG,"-")
830  LOCATE LIGNE,POSITION
840  BY$=INKEY$
850  IF BY$="" GOTO 840
860  IF BY$="0" GOTO 880
870  IF VAL(BY$)=0 AND NUME=1 THEN GOTO 840
880  IF ASC(BY$)=13 THEN GOTO 980
890  IF ASC(BY$)<>8 THEN GOTO 940
900  IF LEN(BZ$)=0 THEN BEEP:GOTO 840
910  LOCATE LIGNE,POSITION+LEN(BZ$)-1:PRINT"-";:BZ$=LEFT$(BZ$,LEN(BZ$)-1)
920  LOCATE LIGNE,POSITION+LEN(BZ$)
930  GOTO 840
940  PRINT BY$;
950  BZ$=BZ$+BY$
960  IF LEN(BZ$)=LONG THEN BEEP:GOTO 980
970  GOTO 840
980  IF MINI=MAXI THEN RETURN
990  VALEUR=VAL(BZ$):IF VALEUR<MINI OR VALEUR>MAXI THEN GOTO 790
1000 RETURN
```

EXPLICATION DU PROGRAMME

lignes 10 à 50 : configuration du PIA. O en sortie et R en entrée,

lignes 60 à 90 : affichage et composition du nombre de zones,

lignes 100 à 190 : composition heure du premier cycle dans A à l'aide du sous-programme d'entrée de 790 à 1000,

lignes 200 à 290 : composition fin de premier cycle dans A1,

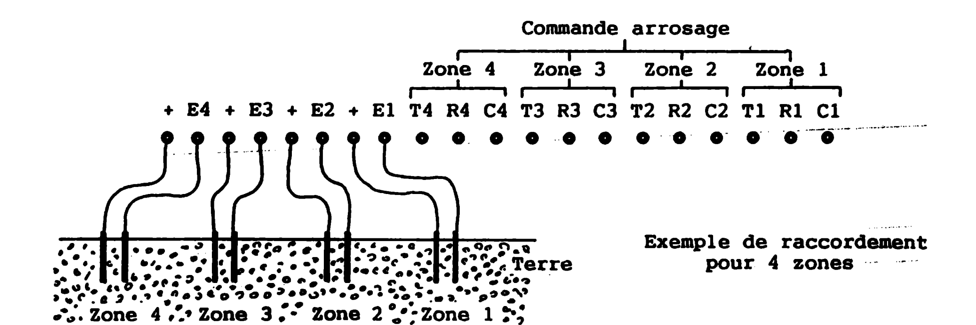
lignes 310 à 400 : composition heure début deuxième cycle dans B,

lignes 410 à 500 : composition heure fin de deuxième cycle dans B1,

lignes 510 à 580 : si entrée " " (barre espace) retour au BASIC. Heure dans C si C supérieur ou égal à A et inférieur à A1 ou C supérieur ou égal à B et inférieur à B1 aller à 610 (heure dans la fourchette cycle 1 ou 2),

lignes 590 à 600 : test des capteurs : si 1 est activé commande sortie 2, si 2 est activé, commande sortie 3, si 3 est activé, commande sortie 4, si 4 activé commande sortie 1 et ainsi de suite en tenant compte (CY) du nombre de zones à arroser,

lignes 790 à 1000 : sous-programme composition des entrées.



Exemple de raccordement pour 4 zones

3. PROGRAMMES ALARM 1

Avec la carte 4 entrées/4 sorties.

PROGRAMME ALARM1

(Voir exemple de raccordement page 201.)

Sous GW BASIC, faire RUN"ALARM1"

Le programme affiche un tableau et un point clignote à côté de "Entrée 1".

Le programme vous demande le numéro de la sortie de la carte 4 entrées/ 4 sorties à commander si l'entrée 1 est activée.

Composer, par exemple, 3 (pas besoin de faire entrée).

Le point clignote maintenant dans la case durée.

Composer la durée pendant laquelle la sortie 3 sera commandée si l'entrée 1 est activée (en composant 0, la sortie 3 restera activée en permanence, en composant 10, elle restera activée pendant 10 secondes, en composant 3 600, pendant 1 heure, avec un maximum de 9 999 secondes) par exemple, composer 25.

Le programme passe à entrée 2. Si l'on compose 0 (entrée 2 à zéro), le programme passe à entrée 3, puis en composant 0, à entrée 4 puis à nouveau 0, le programme passe en alarme. Les entrées et les sorties sont affichées sur un tableau.

Si l'une des entrées est activée (boucle de contact série ouverte ou capteur passant sous son seuil), le relais de la sortie sélectionnée dans le tableau de droite collera.

Au bout du temps programmé en secondes, il décollera même si l'entrée reste activée (il faudra qu'elle soit désactivée, puis activée de nouveau pour faire recoller le relais).

Une sortie sélectionnée dans le tableau de gauche avec son temps à zéro, restera collée en permanence même si l'entrée correspondante activée redevient inactive.

TABLEAU

Si entrée 1 activée : relais *sortie 4* colle et reste collé même si l'entrée 1 se désactive et est activée de nouveau.

Si entrée 2 activée : relais sortie 3 colle pendant 10 secondes et redécolle même si l'entrée 2 reste activée.

Il ne pourra recoller que si l'entrée 2 se désactive, puis redevient active.

Si entrée 3 activée : **relais sortie 4 colle pendant 125 s, puis il se passe le même phénomène que pour entrée 2.**

Si entrée 4 activée : relais sortie 1 colle pendant 9 secondes puis il se passe le même phénomène que pour les deux entrées précédentes.

Entrée n	Sortie n	Durée
1	4	0
2	3	10
3	4	125
4	1	9

Arrêt alarme et retour à la composition tableau en appuyant sur barre espace.

Retour au Basic en appuyant sur F2.

```
10 CLS:KEY OFF:COLOR 7,0
20 PORTA=&H310:PORTB=&H311:OUT &H313,&H82
30 FOR X=1 TO 5:FOR I=1 TO 3:A$((I),(X))="            ":NEXT I:NEXT X
40 NL=5:COLT=3
50 COL(1)=11:COL(2)=8:COL(3)=5:LH=6:DB=36
60 A$((1),(1))="  entrée n ":A$((2),(1))="sortie n ":A$((3),(1))="durée"
70 A$((1),(2))="  entrée 1 ":A$((1),(3))="  entrée 2 ":A$((1),(4))="  entrée 3 "
80 A$((1),(5))="  entrée 4 "
90 KEY(2) ON
100 ON KEY(2) GOSUB 1490
110 LOCATE 25,10:PRINT"        Retour au BASIC appuyer sur touche F2        ";
120 LOCATE 3,40:COLOR 0,7:PRINT" Tableau  commandes ";:GOLOR 7,0
130 GOSUB 460
140 FOR I= 1 TO 4
150 POSI=50:POSI1=59
160 GOSUB 200
170 S1$(I)=SO$:D1$(I)=D1$
180 NEXT I
190 GOTO 760
200 LOCATE 2*I+7,POSI:COLOR 31,0:PRINT"█";
210 COLOR 7,0
220 SO$=INKEY$:IF SO$="" GOTO 220
230 IF SO$="0" THEN D1$="0":LOCATE ,POSI:PRINT"     ":RETURN
240 IF VAL(SO$)>4 OR VAL(SO$)<1 THEN BEEP :GOTO 220
250 LOCATE ,POSI:PRINT"    ";SO$;"  ";
260 LOCATE ,POSI1:COLOR 31,0:PRINT "█    ";:LOCATE ,POSI1:COLOR 7,0
270 D1$=""
280 A$=INKEY$
290 IF A$="" GOTO 280
300 IF A$="0" GOTO 330
310 IF ASC(A$)=13 GOTO 370
320 A=VAL(A$):IF A=0 GOTO 260
330 D1$=D1$+A$
340 IF LEN (D1$)>4 THEN BEEP :GOTO 260
350 PRINT A$;
360 GOTO 280
370 RETURN
380 FOR X=1 TO 4:FOR I=1 TO 4:A$((I),(X))="            ":NEXT I:NEXT X
390 NL=4:COLT=4:COL(1)=10:COL(2)=8:COL(3)=10:COL(4)=8:LH=15:DB=1
400 A$((1),(1))=" entrée 1 ":A$((1),(2))=" entrée 2 ":A$((1),(3))=" entrée 3 "
```

```
410 A$((1),(4))=" entrée 4 "
420 A$((3),(1))=" sortie 1 ":A$((3),(2))=" sortie 2 ":A$((3),(3))=" sortie 3 "
430 A$((3),(4))=" sortie 4 "
440 GOSUB 460
450 GOTO 780
460 '--nl=nombre-ligne--colt=nb-colonne----col(1),col(2),etc-nb-dans-colon-
470 '--lh=ligne-à-partir-du-haut----db=distance-du-bord---dim-col=nbr--colon-
480 IND=0
490 LOCATE LH,DB :PRINT "┌";
500 FOR X=1 TO COLT
510 PRINT STRING$(COL(X),196);
520 IF X=COLT THEN PRINT"┐" ELSE PRINT"┬";
530 NEXT X
540 FOR I= 1 TO NL
550 LOCATE ,DB
560 PRINT "│";
570 FOR X= 1 TO COLT
580 PRINT LEFT$(A$((X),(I)),COL(X));"│";
590 NEXT X
600 PRINT""
610 IF I=NL GOTO 670
620 LOCATE ,DB:PRINT"├";
630 FOR X= 1 TO COLT
640 PRINT STRING$(COL(X),196);
650 IF X=COLT THEN PRINT"┤" ELSE PRINT"┼";
660 NEXT X
670 NEXT I
680 LOCATE ,DB
690 PRINT "└";
700 FOR X=1 TO COLT
710 PRINT STRING$(COL(X),196);
720 IF X=COLT THEN PRINT"┘";ELSE PRINT"┴";
730 NEXT X
740 LOCATE 1,1
750 RETURN
760 LOCATE 10,1
770 GOTO 380
780 NL=1:COLT=1:COL(1)=10:LH=1:DB=1
790 GOSUB 460
800 KEY(2) OFF
810 LOCATE 25,10:PRINT" Retour composition constantes,appuyer barre espace ";
820 LOCATE 12,12:COLOR 0,7:PRINT " ALARME validée ";:COLOR 7,0
830 AA$=TIME$:LOCATE 2,3:PRINT TIME$
840 U=INP(PORTB)
850 IF TIME$<> AA$ GOTO 1360
860 IF INKEY$=" " THEN OUT PORTA,0:COLOR 7,0:CLS:GOTO 30
870 IF(U AND 1)=1 THEN GOTO 960
880 LOCATE 16,13:PRINT"          ";:U1=0
890 IF (U AND 2)=2 THEN GOTO 1060
900 LOCATE 18,13:PRINT"          ";:U2=0
910 IF (U AND 4)=4 THEN GOTO 1160
920 LOCATE 20,13:PRINT"          ";:U3=0
930 IF (U AND 8)=8 THEN GOTO 1260
940 LOCATE 22,13:PRINT"          ";:U4=0
950 GOTO 840
960 LOCATE 16,13:PRINT" activée";
970 IF U1= 0 THEN U1=1 : GOTO 890
980 IF INDS(1)=1 GOTO 1040 ELSE U1=U1+1
990 IF U1>2 THEN U1=4:GOTO 890
1000 IF VAL(S1$(1))>0 THEN SOR(1)=VAL(S1$(1)):CDE(1)=VAL(D1$(1))
1010 IF VAL(S2$(1))>0 THEN SOR1(1)=VAL(S2$(1)):CDE1(1)=VAL(D2$(1))
1020 IF SOR(1) >0 OR SOR1(1) > 0 THEN GOTO 1040 ELSE GOTO 890
1030 INDS(1)=1
1040 LOCATE 14+2*SOR(1),33:COLOR 31,0:PRINT" activée";:COLOR 7,0
1050 GOTO 890
1060 LOCATE 18,13:PRINT" activée";
1070 IF U2=0 THEN U2=1:GOTO 910
1080 IF INDS(2)=1 GOTO 1140 ELSE U2=U2+1
1090 IF U2>2 THEN U2=4:GOTO 910
1100 IF VAL(S1$(2))>0 THEN SOR(2)=VAL(S1$(2)):CDE(2)=VAL(D1$(2))
1110 IF VAL(S2$(2))>0 THEN SOR2(2)=VAL(S2$(2)):CDE2(2)=VAL(D2$(2))
1120 IF SOR(2) >0 OR SOR2(2) > 0 THEN GOTO 1140 ELSE GOTO 910
1130 INDS(2)=1
1140 LOCATE 14+2*SOR(2),33:COLOR 31,0:PRINT" activée";:COLOR 7,0
```

```
1150 GOTO 910
1160 LOCATE 20,13:PRINT" activée";
1170 IF U3=0 THEN U3=1:GOTO 930
1180 IF INDS(3)=1 GOTO 1240 ELSE U3=U3+1
1190 IF U3>2 THEN U3=4:GOTO 930
1200 IF VAL(S1$(3))>0 THEN SOR(3)=VAL(S1$(3)):CDE(3)=VAL(D1$(3))
1210 IF VAL(S2$(3))>0 THEN SOR3(3)=VAL(S2$(3)):CDE3(3)=VAL(D2$(3))
1220 IF SOR(3) >0 OR SOR3(3) > 0 THEN GOTO 1240 ELSE GOTO 930
1230 INDS(3)=1
1240 LOCATE 14+2*SOR(3),33:COLOR 31,0:PRINT" activée";:COLOR 7,0
1250 GOTO 930
1260 LOCATE 22,13:PRINT" activée";
1270 IF U4=0 THEN U4=1:GOTO 840 ELSE U4=U4+1
1280 IF INDS(4)=1 GOTO 1340
1290 IF U4>2 THEN U4=4:GOTO 840
1300 IF VAL(S1$(4))>0 THEN SOR(4)=VAL(S1$(4)):CDE(4)=VAL(D1$(4))
1310 IF VAL(S2$(4))>0 THEN SOR4(4)=VAL(S2$(4)):CDE4(4)=VAL(D2$(4))
1320 IF SOR(4) >0 OR SOR4(4) > 0 THEN GOTO 1340 ELSE GOTO 840
1330 INDS(4)=1
1340 LOCATE 14+2*SOR(4),33:COLOR 31,0:PRINT" activée";:COLOR 7,0
1350 GOTO 840
1360 FOR I=1 TO 4
1370 IF SOR(I)=0 THEN GOTO 1470
1380 IF SOR(I)=4 THEN SSOR=8 :GOTO 1410
1390  IF SOR(I)=3 THEN SSOR=4 :GOTO 1410
1400 SSOR=SOR(I)
1410 CDE(I)=CDE(I)-1 :IF CDE(I)=0 GOTO 1440
1420 TOTSORT=TOTSORT OR SSOR
1430 GOTO 1460
1440 LOCATE 14+SOR(I)*2,33:PRINT"  repos ":INDS(I)=0
1450 SSOR=15-SSOR:TOTSORT=(TOTSORT AND SSOR):SOR(I)=0
1460 OUT PORTA,TOTSORT
1470 NEXT I
1480 GOTO 830
1490 RETURN 1500
1500 COLOR 7,0:CLS:OUT PORTA,0:END
```

PROGRAMME ALARM2

Avec la carte 4 Entrées/4 Sorties + le Mini Clavier
(programme seulement sur la disquette)

Brancher un miniclavier (voir Miniclavier dans dossier PC) sur la prise 2 × 5 broches de la carte PIA.

Le programme ALARM2 est identique à ALARLM1, mais un code secret composé sur le miniclavier permet d'arrêter l'alarme.

Sous GW BASIC faire RUN "ALARM2"

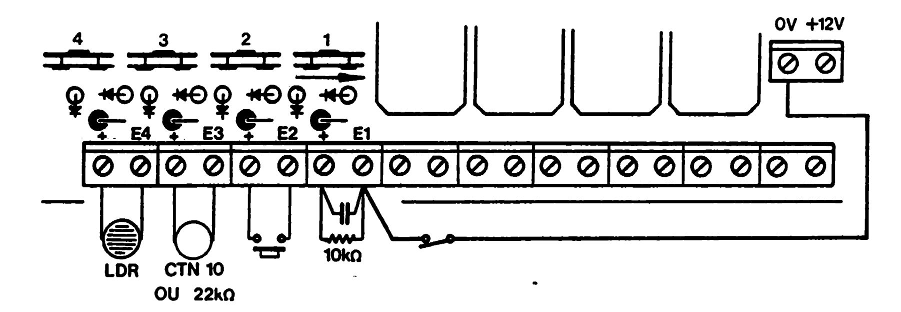
Composer au clavier de l'ordinateur le code secret (maximum 7 chiffres), puis appuyer sur la touche entrée (*enter*). Si code OK, appuyer sur la lettre O, si erreur sur N. Dans le dernier cas, recomposer le code. La suite du programme est identique à ALARM1.

Après la composition des commandes, lorsque l'alarme est validée, la composition du code secret sur le miniclavier suivi d'un appui sur la touche correction de ce même miniclavier arrêtera l'alarme en remettant toutes les sorties au repos. Le programme laisse affiché les sorties qui étaient éventuellement activées et l'heure affichée est celle du moment de l'arrêt de l'alarme. En cas d'erreur de composition du code secret, appuyer sur la touche correction du miniclavier et recomposer le code secret.

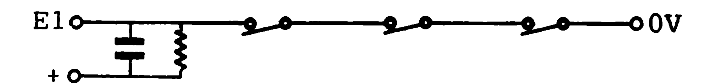
Un appui sur la touche A réactive l'alarme.

En appuyant sur la barre d'espacement, on revient à la composition du code secret puis des commandes.

EXEMPLES DE RACCORDEMENT

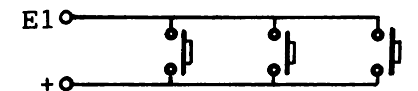


L'entrée E1 sera activée à l'ouverture du contact. On peut monter autant de contacts qu'on le désire (toutes les portes et les fenêtres par exemple) à condition qu'ils soient montés en série entre E1 et le 0 V.



Ne pas oublier la résistance de 10 kΩ entre E1 et le plus et régler l'ajustable 1 à fond comme indiqué sur le schéma de raccordement (résistance maximale). Un condensateur de 0,1 µF sera monté entre E1 et le plus si les longueurs de fils sont importantes.

L'entrée E2 sera activée à la fermeture du contact. On peut monter autant de contacts que l'on veut à condition de les monter en parallèle entre E2 et le plus. On placera aussi un 0,1 µF entre E2 et le plus si les fils sont longs. Régler l'ajustable 2 en position milieu.

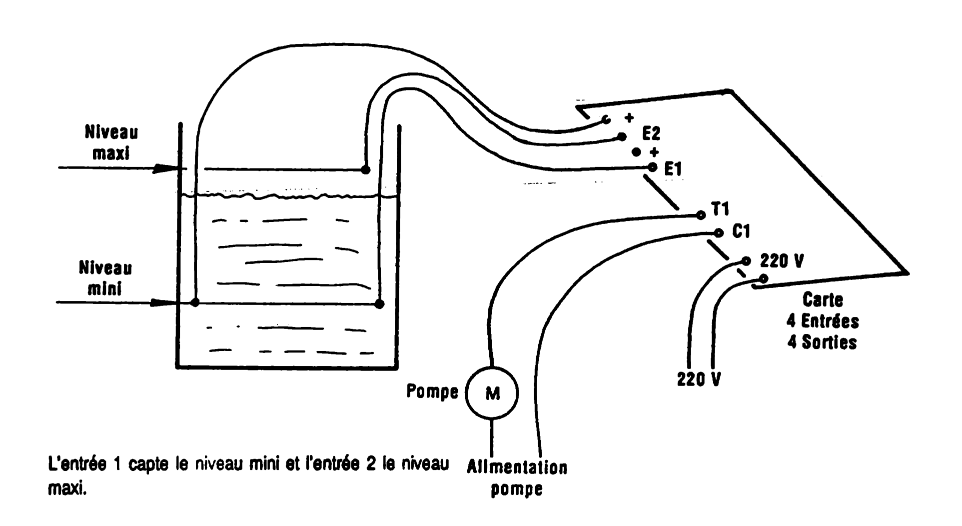


L'entrée E3 sera activée à partir d'un certain seuil de température. Régler l'ajustable 3 pour obtenir le seuil désiré.

L'entrée 4 sera activée si la photorésistance, ou le phototransistor, est éclairé. Régler l'ajustable 4 pour obtenir le seuil de déclenchement en fonction de la luminosité.

4. PROGRAMME RÉSERVE

Composer Run "RESERVE" sous basic



L'entrée 1 capte le niveau mini et l'entrée 2 le niveau maxi.

PROGRAMME RÉSERVE

Ce programme gère le remplissage automatique d'un réservoir.

Lorsqu'on atteint le niveau minimal, la pompe se met en marche. En arrivant au niveau maximal, elle s'arrête.

Il faudra régler les ajustables 1 et 2 de la carte 4E/4S de façon à ce que lorsque les entrées sont dans l'eau, les LED correspondantes (sur la carte 4E/4S) s'allument.

Les capteurs de niveau seront réalisés avec des plaques en inox et reliés par des fils aux points E1, E2 et + de la carte 4 Entrées/4 Sorties.

```
10 '--config--pia,-A--en--sortie,-B--en--entrée
20 PORTA=&H310:PORTB=&H311
30 OUT &H313,&H82
40 '---programme--reserve
50 CLS:KEY OFF:COLOR 0,7
60 LOCATE 11,29:PRINT"  Gestion  réservoir  "
70 LOCATE 25,20:PRINT" Retour au BASIC,appuyer sur barre espace ";
80 COLOR 7,0
```

```
90 LOCATE 13,29:PRINT" entrée 1 niveau mini "
100 LOCATE 14,29:PRINT" entrée 2 niveau maxi "
110 LOCATE 15,26:PRINT" commande  pompe  sortie 1 "
120 U=INP(PORTB) AND 3:LOCATE 20,32:COLOR 7,0
130 IF U=3 THEN OUT PORTA,0:PRINT" pompe arrêtée  "
140 IF U=0 THEN OUT PORTA,1:COLOR 31,0:PRINT" pompe activée  "
150 A$=INKEY$
160 IF A$=" " THEN COLOR 7,0:CLS:OUT PORTA,0:END
170 GOTO 120
```

EXPLICATION DU PROGRAMME

lignes 10 à 30 : configuration du PIA,

lignes 40 à 110 : affichage texte,

ligne 120 : entrée Port B (entrées 1 et 2),

ligne 130 : si mini et maxi=1 arrêt pompe affichage "pompe arrêtée",

ligne 140 : si mini et maxi=0 mise en route pompe (OUT PORTA,1) affichage "pompe activée",

lignes 150 et 160 : test si barre espace appuyée; si oui arrêt programme,

ligne 170 : retour à ligne 120.

5. PROGRAMME TIMERPIA

Avec carte 4 Entrées/4 Sorties

Ce programme permet de commander vingt séquences sur chaque jour de la semaine. Chacune des séquences comporte le choix d'une sortie (1 à 4), l'heure de commande de cette sortie, l'heure d'arrêt de cette sortie.

De plus, vingt séquences sont disponibles pour être exécutées tous les jours de la semaine.

Faire RUN "TIMERPIA" sous BASIC

Le programme vous affiche l'heure et la date et vous demande si l'heure est juste. Si OK, composer le chiffre 0. Si l'heure est fausse, composer le chiffre 1. Dans le dernier cas, recomposer l'heure. L'heure modifiée est affichée de nouveau. Composer 0 si heure OK.

Le programme demande de composer le jour.

ex. si lundi, composer 1. Le programme affiche :

"lundi OK=0 errreur–1"

Si jour OK, appuyer sur chiffre 0 puis entrée (enter). Si erreur, composer –1 et recommencer. Le programme affiche les 20 séquences du lundi et commande 1 en fond inversé.

Vous voulez que chaque lundi, le relais de la sortie 2 colle à 17 h 26 min 30 s et décolle à 17 h 26 min 45 s : composer 2 (pour sortie 2) puis 172 630 ensuite 172 645. Le programme demande d'entrer la commande 2.

Si l'on veut que le relais de la sortie 4 colle à 1 h 25 min 12 s et décolle à 23 h 10 min 15 s : composer 4 (pour sortie 4) puis 012 512 (si l'on ne met pas le 0 devant le 1, l'entrée sera refusée) ensuite 231 015.

Plus de commande pour le lundi, appuyer sur la touche 0.

Le programme affiche les commandes du mardi. Rien pour le mardi, appuyer sur 0 et ainsi de suite jusqu'à la sélection du dimanche.

Le dimanche, on veut commander le relais 2 à 10 h 27 min 30 s et l'arrêter le lundi à 6 h 32 min 00 s : composer 2 puis 102730 ensuite 000000 puis entrée (*enter*).

Commande 2 appuyer sur 0

Le programme affiche la commande pour tous les jours de la semaine.

Appuyer sur 0

Le programme réaffiche les commandes du lundi

Appuyer sur la touche +, la commande 2 est sélectionnée, encore touche +. Sélection commande 3, appuyer sur 2 pour relais 2 puis 000000 ensuite 063200 pour couper le relais 2 le lundi à 6 h 32 min 00 s. Appuyer sur T pour passer à la partie active du programme. Le programme affiche la commande du lundi (si on avait composé jeudi il afficherait jusqu'à 23 h 59 min 59 s la commande du jeudi).

En même temps, il affiche la commande de tous les jours et en bas à gauche l'heure. Lorsque l'heure sera égale à l'heure de l'une des commandes, celle-ci s'effectuera. Au passage de 23 h 59 min 59 s à 00 h l'affichage passera au jour suivant et les données du nouveau jour apparaîtront à l'écran.

En appuyant sur R, on revient à la composition des séquences.

En appuyant sur B, on revient au basic.

Remarque : La date exacte n'a aucune importance pour le programme, seul le jour permet une répétition des commandes semaine après semaine. En cas d'erreur dans la composition d'une commande (heure de collage ou de décollage erronnée), avec la touche + du clavier, aller jusqu'en commande 20.

Appuyer à nouveau sur la touche + pour revenir sur la commande 1 du même jour.

Sélectionner avec la touche + la commande à modifier et recomposer les bonnes heures.

L'appui sur la touche – remet à zéro la commande sélectionnée.

```
5 '--config---pia---
10     CLS:KEY OFF
20     P=&H313:PORTA=&H310:PORTB=&H311
30     OUT P,&H82
40     DIM LC(20):DIM MC(20):DIM YC(20):DIM JC(20):DIM VC(20):DIM SC(20)
50     DIM DC(20):DIM TC(20)
60     DIM LD(20):DIM MD(20):DIM YD(20):DIM JD(20):DIM VD(20):DIM SD(20)
70     DIM DD(20):DIM TD(20)
80     DIM LR(20):DIM MR(20):DIM YR(20):DIM JR(20):DIM VR(20):DIM SR(20)
90     DIM DR(20):DIM TR(20)
95 '--affiche--heure
100    PRINT"Heure=";TIME$;"Jour=";MID$(DATE$,4,3);LEFT$(DATE$,3);RIGHT$(DATE$,2
110    PRINT "Heure ok=0 erreur=1";:INPUT CC:IF CC=0 GOTO 230
115 '--composition--heure---
120    PRINT"composer heure    hh:mm:ss    ";
130    LIGNE=3:POSITION=19:LONG=2:MINI=0:MAXI=23:NUME=1
140   GOSUB 1650
150    A$=BZ$+":"
160    LIGNE=3:POSITION=22:LONG=2:MINI=0:MAXI=59:NUME=1
170   GOSUB 1650
180    BZ$=BZ$+":":A$=A$+BZ$
190    LIGNE=3:POSITION=25:LONG=2:MINI=0:MAXI=59:NUME=1
200   GOSUB 1650
210    A$=A$+BZ$
220    TIME$=A$:CLS:GOTO 100
230    CLS
```

```
235 '---entrée-du--jour--
240   LOCATE 10,30:PRINT "Composer le jour                              "
250   LOCATE 12,20:PRINT " 1=lundi,2=mardi,3=mercredi,4=jeudi"
260   LOCATE 13,22:PRINT " 5=vendredi,6=samedi,7=dimanche "
270   LOCATE 10,48:JO$=INKEY$
280   IF JO$="" GOTO 270 ELSE JOUR=VAL(JO$)
290   IF JOUR <1 OR JOUR >7 THEN GOTO 270
300   J=JOUR:GOSUB 790 :PRINT " ";CSTJOUR$;" ok=0  erreur=-1";:INPUT Z
310   IF Z=0 GOTO 320  ELSE GOTO 230
320 '--entrée--constantes
330   SAUVEJOUR=JOUR
340   CLS
350  FOR J = 1 TO 8 :JOUR=J
360  GOSUB 790:'--constante--jour
370  GOSUB 1460
380   LOCATE 25,7:COLOR 0,7
390   PRINT" Si relais=0 jour suivant,=T Timer,=+ ligne suivante,=- ligne à 0 ";
400   COLOR 7,0
410   LOCATE 1,10 :PRINT "Entrée commande du:";CSTJOUR$
420  FOR I=1 TO 20
430   LOCATE I+3,1:PRINT "commande";:COLOR 0,7: PRINT USING"##";I;
440   LOCATE ,14:COLOR 7,0
450   JR$=INKEY$
460   IF JR$="0" THEN CLS:JR=0:GOTO 770
470   JR=VAL(JR$)
480   IF JR$="-"THEN JR=0 ELSE GOTO 500
490   JC=0:JD=0:GOSUB 880:LOCATE,14:PRINT"0   hh mm ss  hh mm ss ";:GOTO 440
500   IF JR$="+" THEN GOTO 750
510   IF JR$="t" OR JR$="T" THEN JOUR=SAUVEJOUR:J=SAUVEJOUR:GOTO 970:'--timer
520   IF JR<=0 OR JR>4 THEN GOTO 440
530   PRINT USING"#";JR;
540   LOCATE,18:PRINT"hh mm ss";SPACE$(2);"hh mm ss";:JC$=""
550   LIGNE=I+3:POSITION=18:LONG=2:MINI=0:MAXI=23:NUME=1
560  GOSUB 1650
570   JC$=BZ$
580   LIGNE=I+3:POSITION=21:LONG=2:MINI=0:MAXI=59:NUME=1
590  GOSUB 1650
600   JC$=JC$+BZ$
610   LIGNE=I+3:POSITION=24:LONG=2:MINI=0:MAXI=59:NUME=1
620  GOSUB 1650
630   JC$=JC$+BZ$
640   LIGNE=I+3:POSITION=28:LONG=2:MINI=0:MAXI=23:NUME=1
650  GOSUB 1650
660   JD$=BZ$
670   LIGNE=I+3:POSITION=31:LONG=2:MINI=0:MAXI=59:NUME=1
680  GOSUB 1650
690   JD$=JD$+BZ$
700   LIGNE=I+3:POSITION=34:LONG=2:MINI=0:MAXI=59:NUME=1
710  GOSUB 1650
720   JD$=JD$+BZ$
730   JC=VAL(JC$):JD=VAL(JD$)
740  GOSUB 880
750  NEXT I
760   CLS:GOTO 360
770   IF JR=0 AND J=8 THEN GOTO 340
780  NEXT J
790  ON J GOTO 800,810,820,830,840,850,860,870
800   CSTJOUR$="Lundi":RETURN
810   CSTJOUR$="Mardi":RETURN
820   CSTJOUR$="Mercredi":RETURN
830   CSTJOUR$="Jeudi":RETURN
840   CSTJOUR$="Vendredi":RETURN
850   CSTJOUR$="Samedi":RETURN
860   CSTJOUR$="Dimanche":RETURN
870   CSTJOUR$="Tous les jours":RETURN
880  ON J GOTO 890,900,910,920,930,940,950,960
890   LC(I)=JC:LD(I)=JD:LR(I)=JR:RETURN
900   MC(I)=JC:MD(I)=JD:MR(I)=JR:RETURN
910   YC(I)=JC:YD(I)=JD:YR(I)=JR:RETURN
```

```
920  JC(I)=JC:JD(I)=JD:JR(I)=JR:RETURN
930  VC(I)=JC:VD(I)=JD:VR(I)=JR:RETURN
940  SC(I)=JC:SD(I)=JD:SR(I)=JR:RETURN
950  DC(I)=JC:DD(I)=JD:DR(I)=JR:RETURN
960  TC(I)=JC:TD(I)=JD:TR(I)=JR:RETURN
970 '---timer----------
980 GOSUB 1440:'--edit
990 GOSUB 1540
1000  SORT=0
1010  LOCATE 25,22:COLOR 0,7:PRINT" Retour composition=R     Retour BASIC=B ";
1020  COLOR 16,7
1030  TEMPS=VAL(LEFT$(TIME$,2)+MID$(TIME$,4,2)+RIGHT$(TIME$,2))
1040  IF TEMPS=235959! THEN IR=1
1045 '--test--si--jour--change--
1100  IF TEMPS=0 AND IR=1 THEN JOUR=JOUR+1:SAUVEJOUR=JOUR: ELSE GOTO 1140
1110  IF JOUR=8 THEN JOUR=1:SAUVEJOUR=1
1120  IR=0:CLS:COLOR 7,0:GOSUB 1440
1130 GOSUB 1540:COLOR 16,7
1135 '---commande-du-jour--
1140 FOR I=1 TO 20
1150 GOSUB 1350
1155  IF JR=0 THEN GOTO 1210
1160  IF JR=4 THEN JR=8
1170  IF JR=3 THEN JR=4
1180  IF TEMPS=JC THEN SORT=SORT OR JR: OUT PORTA,SORT
1200  IF TEMPS=JD THEN JR=15-JR:SORT=SORT AND JR: OUT PORTA,SORT
1210 NEXT I
1212  LOCATE 25,5:PRINT" ";TIME$;" ";
1220  JOUR1=JOUR:JOUR=8
1225 '--tous--les--jours---
1230 FOR I=1 TO 20
1240 GOSUB 1350
1245  IF JR=0 THEN GOTO 1300
1250  IF JR=4 THEN JR=8
1270  IF JR=3 THEN JR=4
1280  IF TEMPS=JC THEN SORT=SORT OR JR: OUT PORTA,SORT
1290  IF TEMPS=JD THEN JR=15-JR:SORT=SORT AND JR: OUT PORTA,SORT
1300 NEXT I
1310  JOUR=JOUR1
1320  A$=INKEY$:IF A$="R" OR A$="r" THEN SAUVEJOUR=JOUR:COLOR 7,0:GOTO 340
1330  IF A$="b"OR A$="B" THEN COLOR 7,0:CLS:END
1340  GOTO 1030
1345 '--chargement--constantes--
1350 ON JOUR GOTO 1360,1370,1380,1390,1400,1410,1420,1430
1360  JR=LR(I):JC=LC(I):JD=LD(I):RETURN
1370  JR=MR(I):JC=MC(I):JD=MD(I):RETURN
1380  JR=YR(I):JC=YC(I):JD=YD(I):RETURN
1390  JR=JR(I):JC=JC(I):JD=JD(I):RETURN
1400  JR=VR(I):JC=VC(I):JD=VD(I):RETURN
1410  JR=SR(I):JC=SC(I):JD=SD(I):RETURN
1420  JR=DR(I):JC=DC(I):JD=DD(I):RETURN
1430  JR=TR(I):JC=TC(I):JD=TD(I):RETURN
1440  CLS
1450  J=JOUR:GOSUB 790:LOCATE 1,11:PRINT CSTJOUR$
1455 '---tableau---jour--
1460  LOCATE 2,9:COLOR 0,7:PRINT " n° relais   colle   décolle ":COLOR 7,0
1470 FOR I= 1 TO 20
1480 GOSUB 1350
1490  LOCATE I+3,9:PRINT USING"##";I;:PRINT"  ";
1500  PRINT USING"##   ";JR;:PRINT USING"########    ";JC;
1510  PRINT USING"###### ";JD;
1520 NEXT I
1530 RETURN
1535 '---tableau--tous--les--jours--
1540  J=8:GOSUB 790:LOCATE 1,45:PRINT CSTJOUR$
1550  LOCATE 2,43:COLOR 0,7:PRINT " n° relais   colle   décolle ":COLOR 7,0
1560  JOUR1=JOUR :JOUR=8
1570 FOR I= 1 TO 20
1580  GOSUB 1350
```

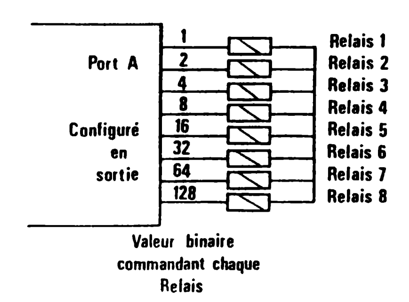
```
1590   LOCATE I+3,44:PRINT USING"##";I;:PRINT"  ";
1600   PRINT USING"##  ";JR;:PRINT USING"########   ";JC;
1610   PRINT USING"###### ";JD;
1620  NEXT I
1630   JOUR=JOUR1 :LOCATE 24,20
1640 RETURN 1010
1650 '---entrée -par -inkeys--long=longueur-enre--mini=valeur-mini
1655 '-maxi=valeur-maxi--ligne=position-pour-locate-résultat--dans--bz$
1660   BZ$=""
1670   LOCATE LIGNE,POSITION:PRINT STRING$(LONG,"-")
1680   LOCATE LIGNE,POSITION
1690   BY$=INKEY$
1700   IF BY$="" GOTO 1690
1710   IF BY$="0" GOTO 1730
1720   IF VAL(BY$)=0 AND NUME=1 THEN GOTO 1690
1730   IF ASC(BY$)=13 THEN GOTO 1830
1740   IF ASC(BY$)<>8 THEN GOTO 1790
1750   IF LEN(BZ$)=0 THEN BEEP:GOTO 1690
1760   LOCATE LIGNE,POSITION+LEN(BZ$)-1:PRINT"-";:BZ$=LEFT$(BZ$,LEN(BZ$)-1)
1770   LOCATE LIGNE,POSITION+LEN(BZ$)
1780   GOTO 1690
1790   PRINT BY$;
1800   BZ$=BZ$+BY$
1810   IF LEN(BZ$)=LONG THEN BEEP:GOTO 1830
1820   GOTO 1690
1830   IF MINI=MAXI THEN RETURN
1840   VALEUR=VAL(BZ$):IF VALEUR<MINI OR VALEUR>MAXI THEN GOTO 1650
1850  RETURN
```

6. PROGRAMME TRUC BASIC

Comment commander les huit sorties d'un port, chacune ayant une fonction indépendante des autres.

Lorsqu'on commande huit sorties, par exemple huit relais, chacun de ces relais doit pouvoir être collé ou décollé sans intervenir sur l'état des autres.

La commande de ces huit relais s'effectuera par exemple sur le port A configuré en sortie (voir figure).



On voit sur la figure que pour faire coller le relais 1, il faudra commander OUT PORT A,1 pour commander les relais 4 et 7 : OUT PORT A,72 (72=8+64), et pour commander tous les relais : OUT PRT A,255 (225=1+2+4+8+16+32+64+128).

EXEMPLE DE PROGRAMME DE COMMANDE DES RELAIS

Sous BASIC. Faire RUN "CDEREL" avec la disquette ou taper le programme suivant :

```
10 'config pia------------
20 CONFIG=&H313
30 PORTA=&H310
40 OUT CONFIG,&H82
50 CLS:KEY OFF
60 SORT=0
70 CLS
80 LOCATE 25,1:PRINT "Retour au BASIC composer -1 "
90 LOCATE 10,20
95 PRINT"cde relais=";SORT;
```

```
100 INPUT" Numéro du relais à coller :";A
110 IF A=-1 THEN CLS:OUT PORTA,0:END
120 IF A<0 OR A>8 THEN BEEP:GOTO 70
130 GOSUB 150
140 GOTO 220
150 Z=1
160 FOR I=1 TO 8
170 IF A=0 THEN RETURN
180 IF A=I THEN A=Z:RETURN
190 Z=Z*2
200 NEXT I
220 SORT =SORT OR A
230 OUT PORTA,SORT
240 CLS
250 LOCATE 25,1:PRINT "Retour au BASIC composer -1 "
260 LOCATE 10,15 :COLOR 0,7
265 PRINT" Cde relais=";SORT;
270 INPUT" Numéro du relais à décoller :";A
275 COLOR 7,0
280 IF A=-1 THEN CLS:OUT PORTA,0:END
290 IF A<0 OR A>8 THEN BEEP:GOTO 260
300 GOSUB 150
310 A=255-A
320 SORT=SORT AND A'
330 OUT PORTA,SORT
340 GOTO 70
```

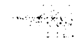
Le programme vous indique :
Cde relais = 0 Numéro des relais à coller?
Retour Basic, composer −1

En composant 4, puis entrée le programme indique :
Cde relais = 8 Numéro du relais à décoller?
8 correspond à la valeur binaire commandant le relais 4

Composer 2, le programme indique :
Cde relais = 8 Numéro du relais à coller?
La valeur binaire n'a pas été modifiée, le relais 2 n'étant pas collé

Composer 1, le programme indique :
Cde relais = 9 Numéro de relais à décoller?
La valeur binaire a été augmentée de 1, valeur binaire commandant le relais 1

Composer 4 :
Cde relais = 1 il n'y a plus que le relais 1 de collé
Retour au Basic, composer −1



EXPLICATION DU PROGRAMME

lignes 10 à 40 : configuration du PIA. Port A en sortie,

lignes 50 à 120 : SORT contient la valeur correspondant à la somme binaire des commandes des relais mis à 0 au démarrage par SORT = 0 (ligne 60) impression de SORT à l'écran (ligne 95) entrée du numéro de relais à commander dans A (ligne 100) test si A=–1 effacement de l'écran arrêt du programme (ligne 110) test si A est entre 0 et 8, sinon émission d'un bip et retour à la ligne 70 (ligne 120),

ligne 130 : aller au sous-programme 150,

ligne 140 : aller à 220,

lignes 150 à 200 : sous-programme permettant la transformation de A (numéro de relais) en sa correspondance binaire, soit :

1, 2, 4, 8, 16, 32 ,64 ,128. Quand A est transformé en binaire, retour soit à 140, soit à 310,

ligne 220 : fonction OU entre A contenant le code du relais à commander et SORT contenant les relais déjà commandés,

ligne 230 : sortie sur PORT A de SORT,

lignes 240 à 275 : gestion de l'écran pour entrer le numéro du relais à décoller,

lignes 280 à 290 : même test qu'aux lignes 110 et 120,

ligne 300 : aller au sous-programme 150,

ligne 310 : complémentation de A à 255 de façon que seule la commande du relais à décoller soit à 0,

ligne 320 : fonction ET entre A contenant le code du relais à décoller et SORT contenant les relais commandés qui vont rester collés,

ligne 330 : sortie sur PORT A de SORT,

ligne 340 : retour à 70.

L'INSTRUCTION OR LIGNE 220

Réalise le OU logique (OR) entre SORT qui contient les relais déjà commandés et A qui contient le code du relais à commander en plus.

Si les relais 1 et 2 sont déjà commandés :

SORT = 3 (en binaire 00000001 + 00000010 = 00000011).

Si l'on veut commander le relais 3 :

A = 4 (après transformation du numéro du relais 3 dans le sous-programme ligne 150 à 200), on a

SORT = 00000011
A = 00000100
00000111 nouveau SORT=SORT OU A.

Après la fonction OU, A a été rajouté à SORT.

Si l'on exécute une deuxième fois le OU logique (OR)

SORT = 00000111
A = 00000100
SORT OU A = 00000111

La valeur de SORT est inchangée, alors que si on avait fait : SORT + A, le nouveau SORT serait égal à 11 (valeur binaire 00001011) et en l'exécutant une troisième fois SORT=11+4=15 et ainsi de suite, alors qu'avec la fonction OU (OR), si le relais est déjà commandé, le nombre résultant du OU logique ne change pas.

Tableau de vérité de la fonction OU (OR)

A	B	C
0	0	0
1	0	1
0	1	1
1	1	1

Fonction C = A ou B

C = 1 si A = 1

C = 1 si B = 1

C = 1 si A = 1 et B = 1

L'INSTRUCTION AND LIGNE 320

ligne 310 : Complémentation de A à 255.

En réalisant A=255–A, on obtient que seule la commande du relais que l'on veut faire décoller soit égal à 0.

Si l'on veut faire décoller le relais 3 (après transformation en ligne 150 à 200 du programme du numéro du relais en sa valeur de commande, soit pour relais 3, valeur de commande A=4) 255–4=251 soit en binaire 251=11111011.

On voit que seule la valeur 4 binaire est à zéro.

Réalisation de la fonction AND (et) entre SORT et A

Soit si SORT=7 (relais 1, 2 et 3 commandés)
SORT=0000111
A=251 =1111011
Nouveau SORT=SORT AND A 0000011

SORT contient l'ancienne valeur 7 diminué de 4 (le relais 3 va donc décoller).

Tableau de vérité de la fonction AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

C n'est égal à 1 que lorsque A et B sont égaux à 1.

En refaisant la même opération sur la nouvelle valeur de SORT

soit SORT=3 00000011
avec A=251 11111011
SORT=SORT AND A 00000011 la valeur reste inchangée.

TEMPORISATION SOUS BASIC

Une temporisation est souvent requise pour la gestion des entrées-sorties. Le moyen le plus simple est de se servir de boucle FOR-NEXT.

Exemple : nous allons déterminer le nombre de fois qu'une boucle FOR-NEXT peut être exécutée en une seconde.

Vérification de la seconde (en effet certains compatibles, Amstrad 1640 entre autre, délivrent une « seconde » qui en fait dure 3 s).

```
10 A$=TIME$
20 IF A$=TIME$ GOTO 20
30 BEEP
40 GOTO 10
```

Composer RUN puis entrée (*enter*) un bip sonore doit retentir toutes les secondes.

Arrêter le programme par les touches CTRL et PAUSE (BREACK), la seconde étant vérifiée.

EXPLICATION DU PROGRAMME

ligne 10 : transfert de l'heure dans A$,

ligne 20 : si l'heure n'a pas changé, on reste ligne 20,

ligne 30 : émission d'un BIP si avancé d'une seconde,

ligne 40 : retour ligne 10.

PROGRAMMES SEC2 ET SEC3

PROGRAMME SEC2

Composer run "SEC2" puis entrée (*enter*).

Ce programme va vous afficher dix nombres correspondants au nombre de boucles à exécuter pour obtenir un délai de une seconde.

```
10 CLS
20 ON TIMER(1) GOSUB 100
30 FOR A=1 TO 10
40 TIMER ON
50 FOR I= 1 TO 50000!
60 NEXT I
70 GOTO 40
80 NEXT A
90 END
100 PRINT I;
110 RETURN 80
```

Ne pas tenir compte du premier nombre qui est souvent erroné. Les neuf autres doivent être sensiblement égaux entre eux. Il faudra diviser l'un de ces neuf nombres par 10 pour obtenir un temps de 0,1 s, par 100 pour obtenir 10 ms.

EXPLICATION DU PROGRAMME

ligne 10 : RAZ écran,

ligne 20 : déclenchement d'une interruption toutes les secondes,

ligne 30 : boucle de dix éditions,

ligne 40 : activation du timer d'interruption,

lignes 50 à 60 : boucle,

ligne 70 : retour à 40 si l'on arrive à faire la boucle 50 000 fois en moins d'une seconde (logiquement impossible),

ligne 80 : +1 dans le compteur d'édition,

ligne 90 : si dix éditions arrêt programme,

ligne 100 : si une seconde, affichage du nombre de boucles en une seconde,

ligne 110 : retour à 80.

PROGRAMME SEC3

Composer RUN"SEC3" puis entrée (*enter*).

```
10 CLS
20 INPUT "nombre/100";TOT
30 FOR A=1 TO 100
40 FOR I= 1 TO TOT
50 NEXT I
60 NEXT A
70 BEEP
80 GOTO 30
```

Composer le nombre obtenu grâce à "SEC2" divisé par 100 (si nombre=1941, composer 19). Un bip sonore doit retentir à peu près toutes les secondes. Arrêter le programme par CTRL et PAUSE.

EXPLICATION DU PROGRAMME

ligne 20 : entrée du nombre dans TOT,

ligne 30 : 100 fois boucle A,

ligne 40 : TOT fois boucle I (=10 ms),

ligne 50 : NEXT I,

ligne 60 : NEXT A,

ligne 70 : bip sonore,

ligne 80 : retour à ligne 30.

On exécute cent fois la boucle de durée = 10 ms puis on commande un bip sonore.

Avec les nombres obtenus, vous pourrez donc créer des boucles de durée inférieure à la seconde.

Imprimerie Arts Graphiques du Perche 28240 Meaucé
Dépôt légal : septembre 1997 — N° d'Imprimeur 971448
Dépôt légal de la 1re édition : juin 1992
Imprimé en France

PC et Robotique

TECHNIQUES D'INTERFAÇAGE

Que vous soyez amateur d'informatique, étudiant ou professionnel de l'automatisme, voici l'ouvrage qui vous permettra d'ouvrir votre ordinateur sur le monde extérieur.

Démarrant avec des exemples d'interfaces très simples, vous évoluerez rapidement vers des montages plus complexes, débouchant sur les problèmes que vous désirez résoudre.

Vous transformerez aisément votre ordinateur en centrale d'alarme multizone, en automate programmable, en voltmètre, en programmateur d'arrosage, etc. Une vingtaine d'applications avec plan du circuit imprimé, schéma, implantation et logiciel associés, vous permettront de mener à bien les réalisations que vous aurez choisies.

La disquette jointe contient tous les programmes décrits dans ce livre.

EDITIONS TECHNIQUES ET SCIENTIFIQUES FRANÇAISES



Code 23883

ISBN 2-85535-201-0

Michel CROQUET

